UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

Máster en Bioinformática y Biología Computacional

TRABAJO FIN DE MÁSTER

COMPARATIVA DE PIPELINES DE ANÁLISIS COMPUTACIONAL PARA LA DETECCIÓN DE TRANSCRITOS CON USO DIFERENCIAL

Autor: Carla Guillén Pingarrón

Tutor: Fátima Sánchez Cabo; Jose Luis Cabrera Alarcón

Ponente: Ramón Díaz Uriarte

Junio 2019

COMPARATIVA DE PIPELINES DE ANÁLISIS COMPUTACIONAL PARA LA DETECCIÓN DE TRANSCRITOS CON USO DIFERENCIAL

Autor: Carla Guillén Pingarrón

Tutor: Fátima Sánchez Cabo; Jose Luis Cabrera Alarcón
Ponente: Ramón Díaz Uriarte

Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio 2019

Resumen

La detección de transcritos que sufren *switching* (cambios en la proporción de isoforma entre distintas condiciones) puede aportar mucha información funcional sobre lo que está ocurriendo en el organismo en un determinado momento: puede que, sin necesidad de que la expresión total del gen varíe o no lo haga en grandes cantidades, se produzca un cambio en la selección de una isoforma respecto a otra. Por este motivo, y debido a la ausencia de un pipeline estandarizado para su análisis, se decidió realizar un *benchmarking* entre las distintas condiciones para comprobar el funcionamiento de cada una de ellas en unas condiciones no limitantes. Para ello, se realizó una simulación de lecturas con el paquete de *Polyester*, de la manera que fueran lo más similares a las biológicas posibles, incluído su variabilidad y sesgos de los métodos de secuenciación más utilizados. Posteriormente, estas lecturas se alinearon mediante dos alineadores especializados en el RNA-seq, STAR y Salmon, que además presentan tiempos de ejecución más cortos que los modelos más antiguos y permiten favorecer estos estudios. La posterior reconstrucción de isoformas y su cuantificación es un paso incluído en Salmon ya que realiza un *quasi-mapeo* de las lecturas y a partir de él realiza la cuantificación de cada isoforma. En el caso del alineador STAR, estas proporciones se estiman mediante RSEM, método que utiliza el algoritmo de esperanza maximización. Por último, para realizar el análisis de la expresión diferencial de porcentajes de isoformas (*switching*) se utilizó DRIMSeq, un paquete de R muy actual que permite la estimación de isoformas; SUPPA, un método utilizado clásicamente para el análisis de cambios en eventos de splicing y ASApp, programa desarollado por la unidad de bioinformática del CNIC. Para poder realizar todos estos pipelines de manera completa, se desarrollaron *scripts* en R y bash que permiten la automatización de cada uno de los pasos y la adaptación de las salidas de uno de los programas para actuar como entrada en la siguiente. Esto, junto al uso del paquete de importación de R *tximport* se obtuvieron los resultados del rendimiento y capacidades de cada uno de los pipelines. Mediante el cálculo de los cambios en la proporción de isoformas calculadas ($\log_2$ Fold Change) para cada uno y la cuantificación del número de lecturas por cada isoforma que nos aporta *Polyester* como las verdaderas (*ground truth*), se pudo calcular el error cuadrado de la media para ver la precisión en la estimación de estas proporciones. Se vio que el mejor paquete en cuanto a desviación era RSEM-AsApp, seguido de Salmon-DRIMSeq. Sin embargo, cuando evaluamos la capacidad de detección de isoformas que sufren *switching* por medio de una curva ROC, vemos que DRIMSeq queda muy por detrás ya que no es capaz de alcanzar grandes valores de sensibilidad, cometiendo una gran cantidad de falsos positivos y falsos negativos. Por otro lado, RSEM-ASApp presenta la mejor curva, concordando con su resultado de MSE, a pesar de la gran cantidad de falsos positivos que introduce en comparación con Salmon-ASApp, debido a que también presenta una gran cantidad de verdaderos negativos, este efecto se ve diluido. Por último, se propone que incrementando la restricción en la determinación de los límites de clasificación de ASApp junto con el pipeline de RSEM podría mejorar el rendimiento conjunto de esta, ya que la estimación de las proporciones sí que son muy acertadas a pesar de que considera como *switching* algunos de los transcritos que no lo sufren.

Palabras clave: splicing alternativo, uso diferencial de transcrito, evaluación comparativa, ARN-seq, bioinformática

Índice

Índice de figuras

1 Introducción

1.1 Motivación del proyecto

La secuenciación masiva genera una gran cantidad de datos que debe ser analizada e interpretada para poder extraer características biológicas relevantes. Muchos laboratorios no poseen un departamento de bioinformática en su centro ni tampoco disponen de personal especializado, siendo incapaces de acceder a esta cantidad de información. Además, existe una gran cantidad de herramientas que realizan las mismas funciones y se incrementan en número cada año, dificultando más la elección del camino a seguir al carecer de comparaciones extensivas. Por ello es importante realizar estudios de benchmarking de las herramientas bioinformáticas existentes para distintos tipos de análisis así como el desarrollo de aplicaciones que cumplan esas guias de buenas prácticas y que puedan ser utilizadas de manera sencilla por usuarios no expertos.

Por otro lado, mientras la secuenciación masiva del ADN ya es una realidad y se realiza en la práctica rutinaria de muchos laboratorios y hospitales, incluyendo en diagnósticos pre-natales o familias con ciertos antecedentes genéticos, el uso del ARN como herramienta diagnóstica se está produciendo de forma progresiva los últimos años. Sin embargo, la secuenciación del ARN o RNA-seq es una de las herramienta clave para entender los mecanismos subyacentes en cualquier condicion biológica, patológica o no. En particular, el splicing alternativo es el responsable de la generación de la variabilidad celular y es esencial en muchos procesos cancerígenos y enfermedades neurodegenerativas [1], [2]. Pese a ello, el análisis de RNA-Seq se ha visto tradicionalmente relegado a la identificación de genes diferencialmente expresados entre condiciones, obviando el análisis a nivel de transcritos que es realmente la unidad transcripcional a nivel biológico. Pese a la relevancia del estudio de la expresión de los transcritos y sus cambios entre condiciones no existe aun un consenso en la comunidad acerca de la herramienta más adecuada, a pesar del uso extendido de unas con respecto a otras, muchas veces sin comprobación formal. La posibilidad de estandarizar un pipeline en función de las condiciones de estudio deseado, así como al final construir una suite que incorpore cada uno de los programas para que funcionen en batería por medio de adaptaciones entre ellos y obteniendo todos los resultados intermediarios para su análisis podría facilitar mucho los estudios y favorecer la comparación entre los distintos experimentos.

1.2 Objetivos y enfoque

El objetivo de este trabajo de fin de Master es evaluar la capacidad para detectar *isoform switch* utilizando una herramienta desarrollada por la unidad de Bioinformática del CNIC (ASApp) en combinación con los alineadores más utilizados actualmente para la cuantificación de transcritos, i.e. STAR y RSEM. Para ello, se compararán dos pipelines de análisis incluyendo ASApp con otros métodos disponibles para la misma tarea, i.e. DRIMSeq y SUPPA. La comparativa se realizará simulando distintas condiciones experimentales en cuanto al numero de reads, réplicas y librería utilizada. Además de esta comparativa para establecer la mejor pipeline de análisis para la deteccion de isoform switch en distintos

contextos experimentales, se pretende implementar una suite donde se puedan ejecutar todos los pasos del análisis: 1) Alineamiento de las lecturas. 2) Reconstrucción de las isoformas y cuantificación. 3) Detección de eventos de *isoform switch* entre condiciones 4) Análisis funcionales y gráficas. Se busca que sea un pipeline realista, que determine el mejor tipo de análisis para las condiciones que se utilizan en la práctica en laboratorios y centros de salud y que ofrezca resultados fácilmente interpretables por el personal sanitario e investigador.

Queremos determinar las condiciones más adecuadas para que la eficiencia de cada una de las herramientas existentes para el estudio del uso diferencial de transcrito sea óptima. Una vez tengamos esas condiciones, el objetivo es poder generar una suite que permita hacer estudios del *switching* de isoformas que pueda utilizarse de forma general en el trabajo de laboratorio por no expertos bioinformáticos intentando aunar las condiciones más frecuentes en los estudios típicos de laboratorio húmedo y los programas con la mejor precisión y eficacia (en tiempos de computación y uso de memoria, para que no se necesiten ordenadores especialmente potentes en el uso de esta suite).

El objetivo final es favorecer los estudios del uso diferencial de transcrito y el splicing alternativo e intentar que se puedan introducir como rutina en el laboratorio de la misma forma que la secuenciación del ADN o la expresión génica, pudiéndose así obtener nueva información para intentar resolver todas las incógnitas y dualidades que se enfrentan en la actualidad y que están descritas en la introducción.

1.3 Metodología y plan de trabajo

Para la consecución de los objetivos anteriores se estableció el siguiente plan de trabajo:

- Estudio crítico de los métodos existentes para el análisis de la expresión de isoformas y de la mejor forma de combinarlos con alineadores produciendo pipelines de análisis (Figura 1).
- Generación de lecturas sintéticas independientes del modelo y representativas de un experimento de RNA-Seq para poder testar cada uno de los pipelines en distintas condiciones experimentales.
- Adaptación y elaboración de scripts para automatizar los pipelines escogidos.
- Evaluación de los pipelines en los distintos contextos en base a estadísticas que evalúen la performance de los métodos.

Además, se propusieron las siguientes mejoras en las funcionalidades de ASApp:

- Incorporación a ASApp de información funcional adicional.
- Adición de gráficos informativos sobre *switching* como puede ser *ribbonplot*.

a) Investigación y búsqueda bibliográfica entre los métodos actuales de análisis de splicing alternativo y determinación de los pipelines más adecuadas que van a testarse en el estudio.
    a. Mapeo de las lecturas: Elección de tres métodos que utilizan distintas aproximaciones para realizar el mapeo de las lecturas
        i. STAR: Método muy rápido para el análisis de las lecturas pero que consume bastante memoria. Para alinear las lecturas, busca el Prefijo Mapeable Maximal (MMP) coincidente entre las lecturas o pares de lecturas y el genoma, utilizando una matriz de indexación de sufijos. El MMP se define como la subsecuencia más larga que mapea exactamente con una o más subsecuencias del genoma de referencia. Dentro de una misma lectura, se

pueden mapear distintas partes en diferentes posiciones genómicas en función del splicing o fusiones en el ARN[3].

ii. Salmon: Realiza un pseudo-alineamiento al mismo tiempo que cuantifica las isoformas. Necesita un transcriptoma de referencia que será indexado para poder asociarlo con la expresión de cada lectura. En nuestro caso esto no será un problema ya que trabajaremos una especia muy conocida como es el humano. Hace la cuantificación de cada isoforma sin realizar el mapeo *per sé*. Tiene en cuenta el sesgo de GC[4].

b. Reconstrucción de isoformas y cuantificación:
   i. RSEM: No requiere un genoma de referencia si das un set de transcritos para que sean esta. Utiliza el algoritmo de esperanza-maximización (EM). Es capaz de trabajar con lecturas que mapean de forma ambigua en varios sitios del genoma.[5]
   ii. Salmon[4]
c. Análisis de expresión diferencial:
   i. DRIMSeq: Permite hacer estimaciones robustas con menos replicados. Como un gen tiene varias isoformas, utiliza un modelo de expresión multivariada de transcritos, la distribución de Diritchlet.[6]
   ii. ASApp: Utiliza un modelo de clasificación mezcla de gaussianas (GMM) para modelar la expresión en cada una de las condiciones y determina la probabilidad de cada elemento de pertenecer al grupo, obteniendo así la posibilidad de establecer un cutoff para la clasificación de *switching.* Para evaluar la expresión diferencial utiliza también como métrica la distancia de Jensen-Shannon. La divergencia de Jensen-Shannon mide la similitud entre dos distribuciones de probabilidad y se utiliza como la métrica la raíz cuadrada de la distancia misma[3]. Permite ajustar el límite de decisión entre las gaussianas.
   iii. SUPPA2: Utiliza una distribución empírica y calcula la diferencia en la proporción entre condiciones y las diferencias en la proporción entre replicados biológicos[7].

b) Desarrollo de unas lecturas sintéticas con el uso de Polyester[8], utilizando un sesgo realista de GC basado en humanos[9] y utilizando distinto grado de *switching* en las isoformas (*fold change*)[10]. Elaboración de un programa que permita la automatización en la generación de lecturas de manera automática para permitir la reproducción de este estudio y la introducción de nuevas condiciones de profundidades.
c) Evaluación de los distintos métodos de splicing alternativo con cada una de las condiciones descritas anteriormente. Se evaluará:
   a. Precisión para cada una de las condiciones, dando más valor a las condiciones que suelen utilizarse de forma común en los laboratorios en la actualidad
   b. Capacidad de detección y número de errores cometidos
   c. Tiempo de computación
   d. Requerimientos de memoria
   e. Interfaz de usuario y facilidad de comprensión y uso

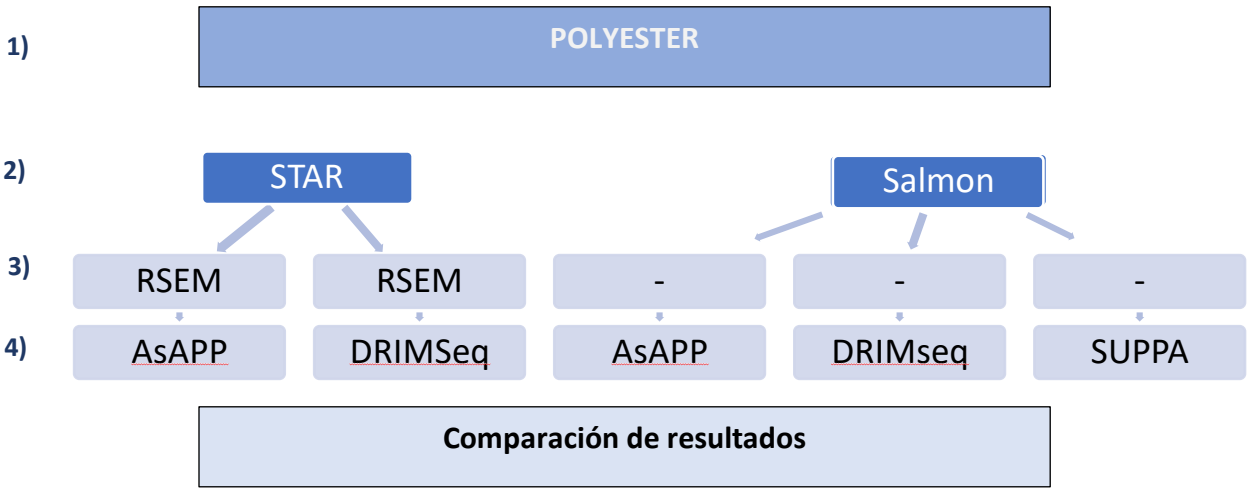


Figura 1. Resumen del protocolo a elegir para testar las diferentes condiciones escogidas. 1) Con *Polyester* diseñamos las lecturas para las condiciones, con el número de replicados y condiciones deseadas, que se testarán de igual manera para cada uno de los pipelines. Posteriormente, estas lecturas van a introducirse en cada uno de los caminos para analizar el splicing alternativo y el uso diferencial de transcrito (DTU). Se describe la herramienta que se utilizará en cada una de las etapas: 2) Mapeo de las lecturas respecto al genoma de referencia (Salida: Bam/Sam), 3) Reconstrucción de las isoformas y cuantificación (Salida: TPM/FPKM), 4) Detección de cambios en los porcentajes de las isoformas. Por último, se realizará una comparación entre las capacidades de cada uno de ellos.

Tareas / subtareas	**Horas**
T1. Diseño y desarrollo del proyecto	**60**
T1.1 Estudio de la bibliografía actual	20
T1.2 Búsqueda de las lecturas sintéticas y biológicas	15
T1.3 Decisión de los programas para los pipelines y su funcionamiento	15
T1.4 Decisión de las variables de los pipelines	10
T2. Generación de las lecturas sintéticas	**60**
T2.1 Manejo del programa de las lecturas sintéticas	10
T2.2 Generación de las variables (%GCs, coverage, fold changes)	15
T2.3 Obtención de las lecturas	15
T2.4 Elaboración de un script para generar lecturas sintéticas	20
T3. Comparación de eficiencia entre pipelines	**100**
T3.1 Correr los distintos pipelines	40
T3.2 Obtención de resultados	10
T3.3 Adaptación de los pipelines: modificar las salidas de un programa para que sirvan como entrada del siguiente	20

T3.4 Obtención de datos para la evaluación (FP, FN)	20
T3.5 Producir gráficas comparativas entre métodos	10
T4. Selección y elaboración de la suite	**10**
T4.1 Estudio de las condiciones más utilizadas en laboratorio	10
T5. Escritura del proyecto	**60**
T5.1 Desarrollo de las imágenes asociadas al proyecto	10
T5.2 Redacción del proyecto	50
T6. Desarrollo de la presentación y defensa del proyecto	**10**
TOTAL HORAS	**300**

2 Estado del arte

2.1 Transcriptómica

Desde la secuenciación del genoma humano en 2001 y gracias a la mejora de todas las técnicas de secuenciación existentes, se ha obtenido una gran cantidad de datos con los que se ha podido estudiar cómo extraer toda la información codificada en forma de esa gran masa de datos. Sin embargo, la complejidad biológica del ADN es simplemente una de las capas de esa información. Aunque todas las células de nuestro cuerpo presentan el mismo ADN (a excepción de las diferentes mutaciones puntuales que puedan adquirir durante su replicación y debido al estrés oxidativo o las condiciones a las que se vean sometidas durante su periodo vital), las diferencias obvias entre un tipo celular y otro se deben a que no todas expresan las mismas regiones de este. El análisis del transcriptoma (regiones del ADN que se están expresando) es una herramienta muy útil para comprender un poco más a fondo la complejidad existente y acercarnos un poco más a entender las diferencias fenotípicas subyacentes. No solo podemos observar las diferencias entre tipos celulares, sino también temporales, entre condiciones…, pudiendo usarse incluso para clasificar tumores entre diferentes fenotipos[11]. La transcriptómica tiene gran relevancia en la caracterización funcional y anotación del genoma: nos permite reconstruir las redes de interacción para comprender distintos sistemas biológicos y establecer determinadas condiciones de expresión que son la firma genética de una enfermedad y con esto ayudar a su diagnóstico precoz [12]; así como a comprender sus bases moleculares, pudiendo llevar a un tratamiento eficaz[1]. Además, la regulación de la expresión génica es la forma que tiene el organismo para adaptarse a los cambios en el ambiente[13], por lo que nos proporciona información sobre ecología molecular.

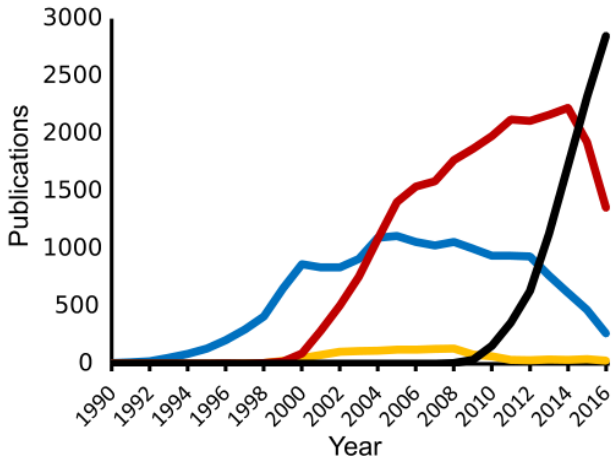


Figura 2. Evolución de los métodos de transcriptómica a lo largo de los últimos años. Publicaciones relacionadas con marcador de secuencia expresada (EST) en azul, análisis de la expresión de los genes en serie (SAGE) en amarillo, microarrays de ADN en rojo y RNA-seq en negro. Lowe, Rohan et al. "Transcriptomics technologies." PLoS computational biologyvol. 13,5 e1005457. 18 May. 2017,doi:10.1371/journal.pcbi.1005457

2.1.2 Técnicas para la transcriptómica

Antes de que el estudio de la expresión resurgiese como una -ómica (estudiar la expresión en su totalidad, como conjunto), ya en 1971 empezaron a obtener ARNs mensajeros (ARNm) y se almacenaron como ADNc. En 1980, con la aparición de la secuenciación Sanger, se empezaron a obtener la secuencia de algunos de estos. También se utilizaba la retro-transcripción sumada a la PCR cuantitativa, pero eran métodos muy complejos y solo eran capaz de obtener una pequeña porción del transcriptómica[14]. Hace 25 años los laboratorios ya clonaban los genes y observaban sus niveles de expresión, intentando caracterizar su función y localizarlo en una vía particular. El objetivo era obtener la mayor cantidad posible de información partiendo de datos. Mucho antes de la publicación del genoma humano completo, ya estaban en desarrollo muchas técnicas para permitir la identificación de regiones expresadas y genes, pudiendo realizar una anotación del genoma para identificar distintas regiones funcionales y reguladoras. En 1991 se publicó el primer intento de obtención de transcriptoma, llegando a obtener la secuencia de más de 600 genes expresándose en el cerebro, pudiendo identificar 337 nuevos con el uso de la tecnología de marcador de secuencia expresada[15]. El gran interés hizo que se dedicasen muchos recursos a desarrollar nuevas tecnologías e intentar mejorar la sensibilidad de los métodos ya existentes. Con la aparición de la secuenciación masiva, el paradigma cambia: trabajamos con muchos genes a la vez y por ello muchos transcritos, permitiendo estudiar las interacciones y redes de interacción y analizar así la expresión de todo el transcriptoma. La transcriptómica es una rama que se caracteriza por el continuo progreso gracias a los desarrollos producidos tanto a nivel técnico-biológico como informático. Esta evolución puede observarse en la Figura 2 y describirse como:

- **Marcador de la secuencia expresada**: Esta técnica se basa en la purificación de ARNm a partir de su cola de poli-A. Posteriormente se hace una transcripción reversa para obtener la versión de ADNc o ADN complementario, creando una librería. Luego, estos fragmentos se amplifican utilizando como cebador para la replicación pequeños *primers* universales que se pueden asociar a distintas regiones del ADNc para su secuenciación e identificación, obteniendo las bases que componen esa región del gen expresado (EST) y permitiendo localizarla en el genoma gracias a las herramientas bioinformáticas, actuando como "marcador de secuencia expresada"[16]. En la época en la que la secuenciación de todo el genoma era algo inviable para realizar en la práctica rutinaria, fueron la primera herramienta utilizada. Se utilizó ampliamente durante muchos años (Figura 2Figura 2) debido a su simplicidad, siendo además muy barata y rápida debido a la automatización de todos los procesos[17]. Aunque pueden utilizarse para la obtención de perfiles transcripcionales (la abundancia relativa del gen se define como la razón entre el EST homólogo y el número total de ESTs), esta función ha quedado relegada a otras técnicas más modernas que presentan mayor precisión. Aun así, sigue siendo fundamental para la anotación del genoma y descripción de nuevos genes[18], [19].

- **SAGE**: En 1995 se utilizó la técnica de análisis de la expresión génica en serie (SAGE) para realizar el primer análisis del transcriptoma. Utiliza la secuenciación de Sanger sobre ADNc purificados por las colas de poli-A, que posteriormente son cortados por una enzima de restricción, dejando extremos cohesivos que permiten el anillamiento de unos adaptadores, cortando en extremo romo y permitiendo la amplificación por PCR para posteriormente eliminar los adaptadores y unir todos los pequeños tags de ARNm que han quedado en una gran secuencia; esta se

secuenciará permitiendo detectar la expresión (Figura 3). Esta técnica es más sensible para transcritos que se expresaban en poca cantidad ya que el método de EST utiliza la secuenciación de un único transcrito; por lo que cada secuencia representa un transcrito único. Sin embargo, SAGE tiene múltiples tags para cada transcrito, y además una única secuencia de tag es la unión de múltiples transcritos: cada secuencia de SAGE representa varios genes y cada gen aparece más de una vez en la misma, permitiendo la identificación de más ARNs y a su vez detectar mejor los transcritos que están en poca abundancia [20]. Así, se empezó a utilizar esta técnica como sustitutiva para medir la expresión, dejando la EST para la identificación de nuevos transcritos.

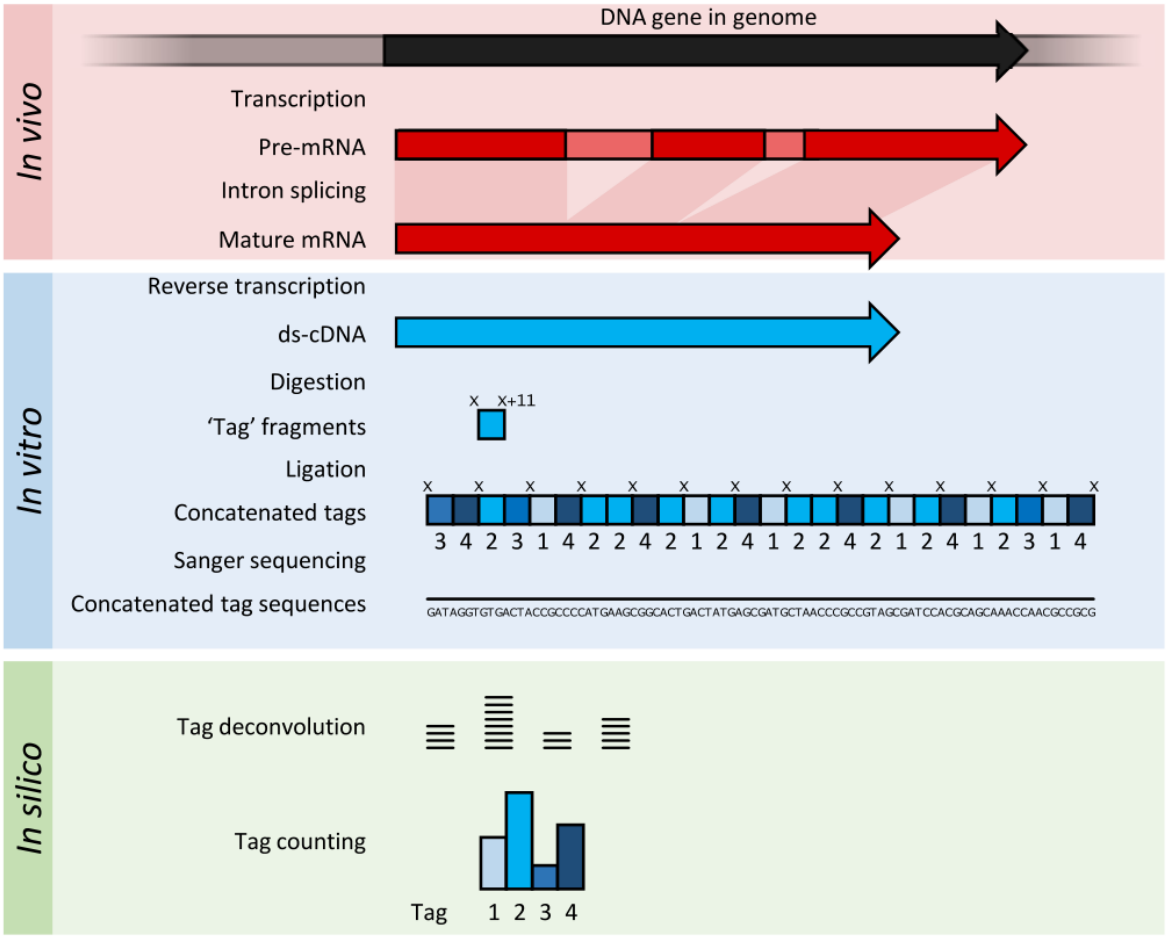


Figura 3. Resumen de la técnica de SAGE. Dentro del organismo, el ADN que se está expresando se transcribe, generando un Pre-ARNm, que madura por medio del splicing y genera un ARNm que presenta un cap en el 5' y una cola de poliA en el 3'. A partir de estos se capturan los ARNm maduros, se genera un ADNc y mediante la digestión obtenemos los tags, que posteriormente se ligarán para dar lugar a una concatenación, que se secuencia. Después, se produce una deconvolución de la misma para separarlo en los distintos tags de origen. La frecuencia de cada tag se utiliza como medida de la expresión del gen del que provienen.

- **Microarrays:** Gracias a la anotación de genes que permitió la tecnología EST se pudo dar paso a los microarrays, que están formados por una plataforma sólida que tiene unidas sondas (obtenidas gracias a tecnología como el EST) para hibridar con nuestro ARN. Podemos determinar la expresión de cada transcrito, marcado por fluorescencia, por medio de la

hibridación con las sondas y detectando la intensidad. Permite detectar de forma simultánea la expresión de multitud de genes (en función del array escogido). Su introducción fue una mejora a nivel de detección de la expresión de forma masiva (ómica), pero se acompañó de un incremento en la complejidad de análisis, requiriendo un mayor conocimiento computacional que los métodos anteriores, así como cambios en las estadísticas realizadas. Además, adicionalmente al gran volumen de información generado, presentan mucha variabilidad, por lo que el diseño experimental y el análisis matemático son claves en estos. Un microarray es una plataforma sólida que presenta sondas con la secuencia complementaria a distintos transcritos conocidos del organismo para que se produzca una hibridación. Pueden describirse dos tipos:

- **De un color:** Muestra y control hibridan de manera independientemente en dos *arrays* (hibridación no competitiva). Hay un gradiente de colores directamente proporcional a la intensidad de la hibridación: las zonas blancas son zonas de mucha expresión mientras que los colores oscuros representan una expresión menor. Se comparan las dos imágenes de los *arrays* para ver si hay algún cambio entre condiciones
- **Dos colores:** Caso y control se marcan de distinto color (ej. Rojo de condición uno y verde en condición 2) y se hibridan de forma competitiva sobre la misma plataforma. Los puntos donde la expresión del gen aparece totalmente verde es un gen que no se expresa en la condición 1 pero sí en la condición dos, si hibrida ocurriría lo contrario. La mezcla de ambos es amarilla. Realmente se obtienen gradientes, donde en función del color predominante sabremos si la expresión de ese gen está sobreexpresado o infraexpresado en una condición u otra.

Independientemente del tipo, el resultado que se obtiene tras digitalizar las intensidades de las imágenes una matriz numérica con los genes en las filas y los datos de experimentos en columnas.

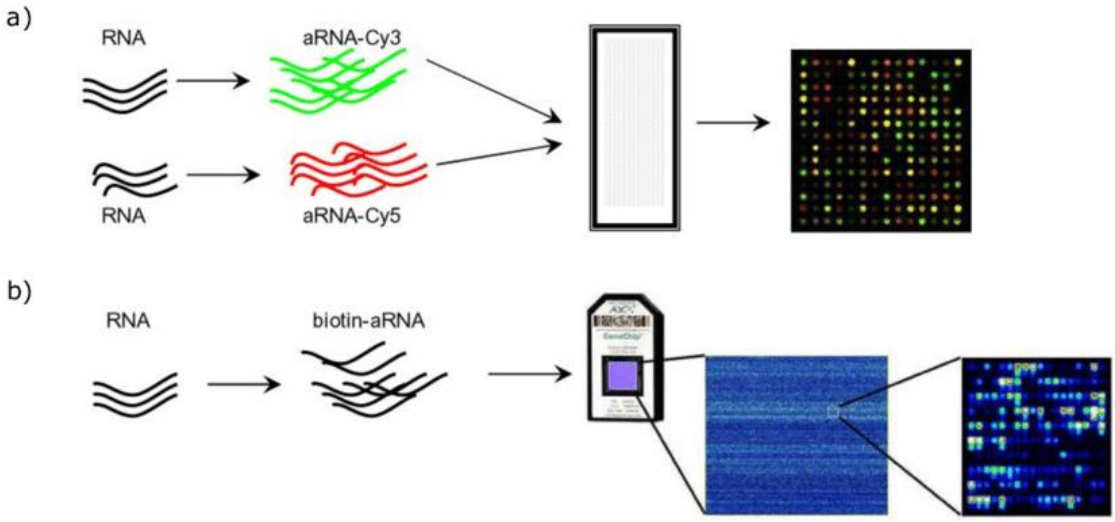


Figura 4. Vista esquemática de los dos tipos de microarrays utilizados por Affymetrix, a) Microarray de dos colores y b) array de un solo color.
F. H. J. van Tienen, C. J. H. van der Kallen, P. J. Lindsey, R. J. Wanders, M. M. van Greevenbroek, and H. J. M. Smeets, "Preadipocytes of type 2 diabetes subjects display an intrinsic gene expression profile of decreased differentiation capacity," Int. J. Obes., vol. 35, no. 9, pp. 1154–1164, Sep. 2011.

Esta técnica tiene una gran cantidad de ventajas y desplazó rápidamente a los EST y SAGE en la detección de la expresión génica, sin olvidar que los microarrays parten de la ventaja de las colecciones de EST preexistentes y de todos los datos de secuenciación. Es una técnica muy rápida, con un formato cómodo (tamaño similar a un porta) y con un coste relativo bastante barato. Sin embargo, los arrays suelen incluir solo determinados sets de genes, y el estudio de un transcriptoma completo con arrays sale bastante más caro. Los inconvenientes radican en la mayor complejidad del análisis, que no presentan gran resolución (se satura a niveles altos de señal y con poca señal no se diferencia del fondo), y algunas limitaciones técnicas como hibridación cruzada.

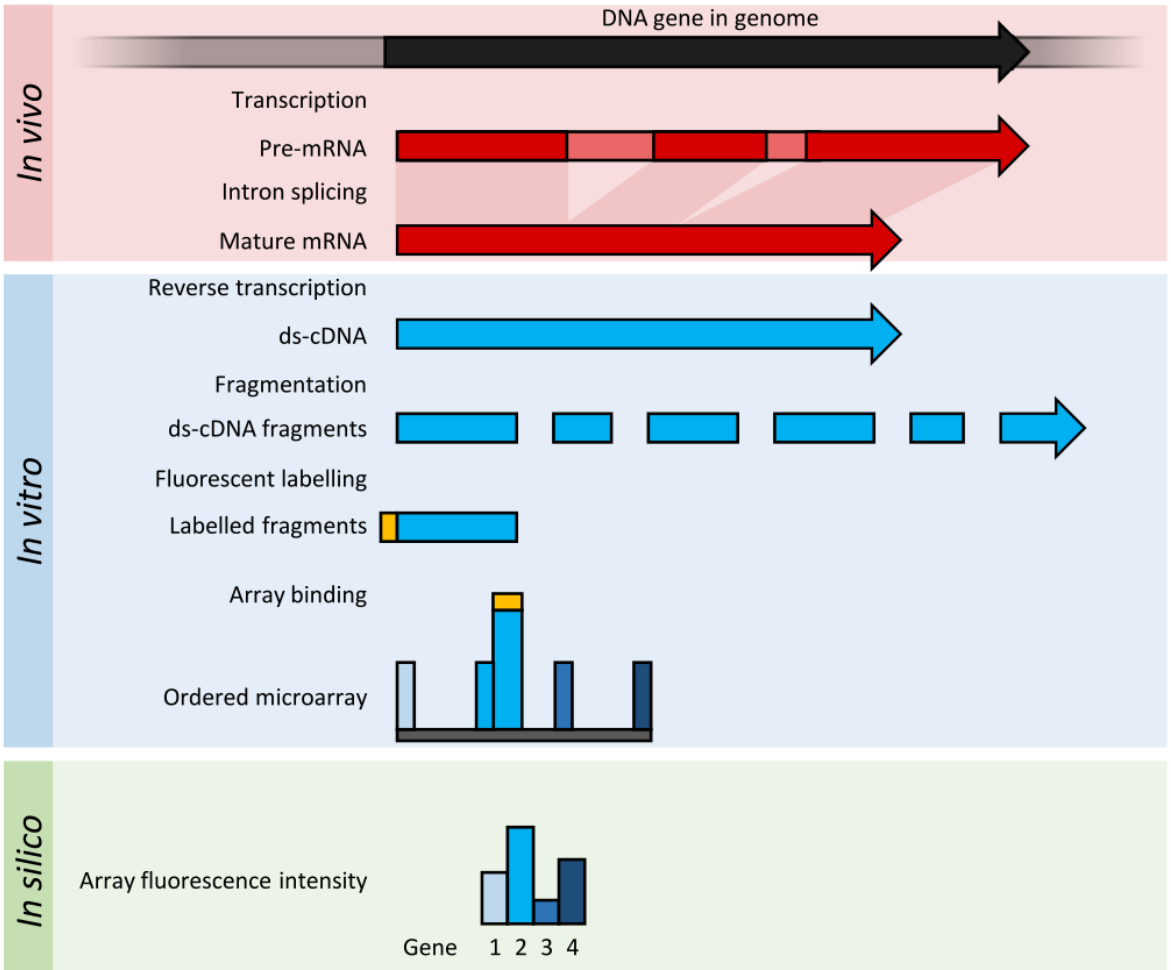


Figura 5. Resumen de la técnica de Microarrays. Dentro del organismo, el ADN que se está expresando se transcribe, generando un Pre-ARNm, que madura por medio del splicing. Se captura el ARNm maduro, se realiza una retrotranscripcción, y tras fragmentar la amplificación, se añade unas etiquetas marcadas (fluorescencia o radiactividad) que permitirá detectar la actividad de ese transcrito. Posteriormente, se disponen para la hibridación en una matriz de microarrays y se observa la expresión de cada uno de los transcritos presentes en el mismo.

- **RNA-seq:** Técnica de perfilado del transcriptoma por medio de tecnologías de secuenciación profunda. Un conjunto de ARNs (total o seleccionando un tipo concreto, como los que sean polyA positivos) se convierten en una librería de fragmentos de ADNc con adaptadores en ambos extremos. Después, cada molécula se secuencia para obtener pequeñas secuencias (con longitudes entre 30-400pb en función de la tecnología utilizada) desde un extremo (single-end) o ambos extremos (paired-end). Combinación de la identificación de la secuencia de los transcritos (EST) y de la detección de los niveles de expresión (en sustitución de los microarray). Un mismo DNA puede dar lugar a distintas isoformas maduras del RNA una vez procesado. Con el RNA-seq podremos determinar la proporción de cada uno o mirar la expresión de forma general a nivel de gen. Gracias a la mejora de los algoritmos y la capacidad de los ordenadores, las -ómicas se han podido estudiar de una forma mucho más extensiva, permitiendo que la transcriptómica pudiese avanzar rápidamente de la mano de los microarrays y el RNA-seq. Debido al desarrollo de las técnicas de secuenciación masiva en los últimos años (NGS, *next generation sequencing*), que es utilizada directamente por el RNA-seq para obtener la secuencia de cada uno de los transcritos, el RNA-seq comenzó a ganar terreno respecto a los microarrays. Con estas técnicas de secuenciación se conseguían eliminar los cuellos de botella al automatizar totalmente la lectura de los resultados y sin la necesidad de utilizar complejos sistemas de separación de las bases.[21]

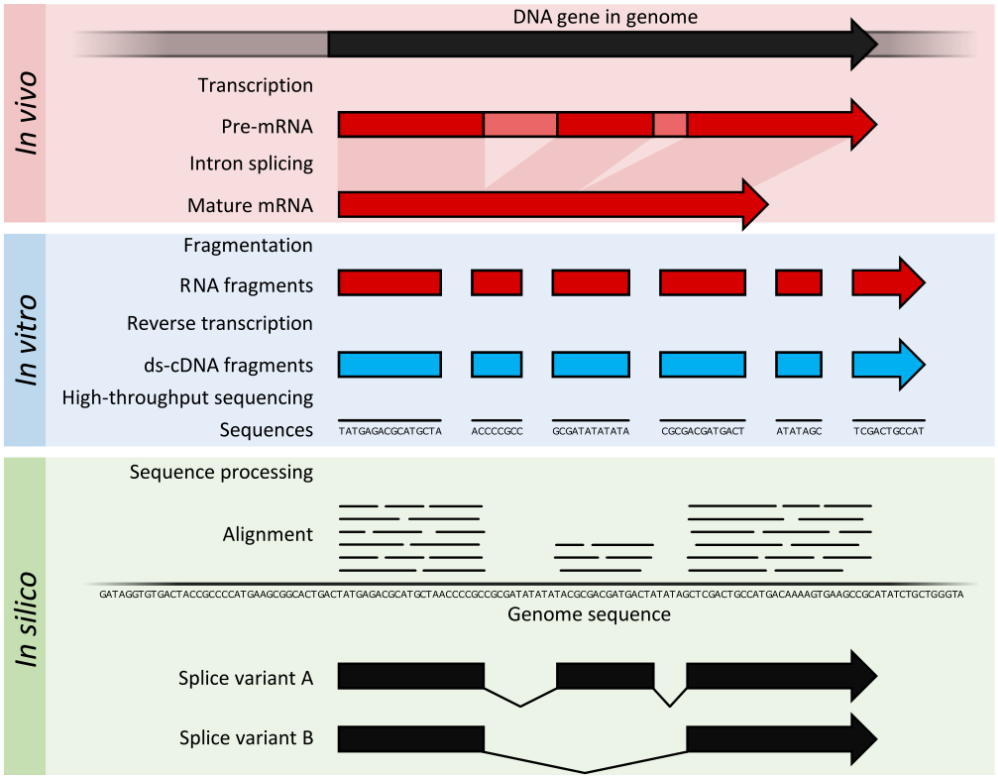


Figura 6. Resumen de la técnica de RNA-seq. Dentro del organismo, el ADN que se está expresando se transcribe, generando un Pre-ARNm, que madura por medio del splicing. Se captura el ARNm maduro, que se fragmenta y se retrotranscribe para obtener ADNc que es secuenciado por medio de las técnicas de secuenciación masiva. Luego, gracias a un procesamiento bioinformático bastante complejo, se alinea contra el ADN molde y obtenemos las diferentes secuencias de transcritos presentes en la muestra.

2.1.2.1 Diferencias entre microarrays y RNA-seq

Inicialmente se utilizaba de forma generalizada la hibridación con *microarrays* ya que su interpretación era mucho más sencilla, sin embargo, la capacidad expresiva estaba muy limitada y tenía grandes problemas de fondo, sensibilidad y saturación. Gracias a la técnica del RNA-seq actualmente podemos obtener el transcriptoma completo sin sesgos ni limitaciones utilizando técnicas de secuenciación de alto rendimiento en vez de observar la expresión de unos determinados genes seleccionados mediante sondas. Este método tiene una mayor sensibilidad al poderse amplificar por PCR. Sin embargo, una de las mayores limitaciones es la dificultad que representa la interpretación de los resultados obtenidos y el procesado de los mismos para obtener la expresión [22].

A pesar de las grandes ventajas que presenta este método, actualmente en muchos laboratorios ha sido imposible desplazar a la técnica de microarrays para la detección de la expresión génica. Entre las causas destaca la necesidad de elección en el RNA-seq de distintos parámetros que pueden incrementar el coste de todo el proceso. Uno de ellos es la cobertura de la secuencia o porcentaje de transcritos cubiertos por el estudio. Una mayor cobertura requiere una mayor profundidad de la secuenciación, con las correspondientes implicaciones en el precio. Además, se pueden seleccionar distintos tamaños para estas lecturas, elegir si hacerlas apareadas o con solo un origen (single-end), etcétera. Dadas las ventajas aportadas por esta tecnología nos centraremos en valorar diversos pipelines basados en los datos de RNA-seq.

Technology	RNA-Seq	Expression microarray	cDNA or EST sequencing
Detection methods	High-throughput sequencing	DNA hybridization	Sanger sequencing
Reliance on reference genome	In some cases	Yes	No
Resolution	Single nucleotide	Probe length (from several to 100 bp)	Single nucleotide
Throughput	High	High	Low
Background noise	Low	High	Low
Cost for mapping transcriptomes of large genomes	Relatively low	High	High
Comparison of gene expression	Counts of reads	Relative intensities	Limited for gene expression
Identification of novel gene and isoform	Yes	No	Yes
SNP detection in the transcribed regions	Yes	No	Yes
Ability to distinguish allelic expression	Yes	No	Yes
Dynamic range of expression levels	>8000-fold	One hundred to a few hundred-fold	Not practical

Figura 7. Ventajas del método de RNA-Seq en comparación con otros métodos transcriptómicos**.**

2.1.2.2 Procesamiento bioinformático

La mayor limitación del RNA-seq sigue siendo el procesamiento de los resultados y el uso de herramientas que requieren cierto conocimiento bioinformático. Los procesos necesarios a seguir hasta obtener un resultado de expresión son los siguientes:

- **Análisis de imagen y llamada de las bases** (*base calling*): Varía en función de la técnica de secuenciación escogida. La secuenciación más extendida en la práctica de los laboratorios es la de *Illumina*. Después de fragmentar el ADN y la adición de los adaptadores a los extremos, realizaremos una amplificación por PCR en una superficie sólida que presenta unas placas (*flowcell*) de cristal donde se han fijado los cebadores. Estas placas tienen ocho canales y pueden secuenciar hasta ocho moléculas en paralelo. Se adiciona a la placa una solución muy diluida del ADNc de manera que se unan de forma dispersa por la placa. Después, se lanza la amplificación por PCR. Como el ADN es flexible, una vez se ha sintetizado la nueva cadena, como solo se encuentra unida por un extremo y por movimiento térmico estas moléculas se pliegan, uniéndose por el otro extremo a otro cebador libre, formando puentes. Esto va creciendo poco a poco en colonias independientes.

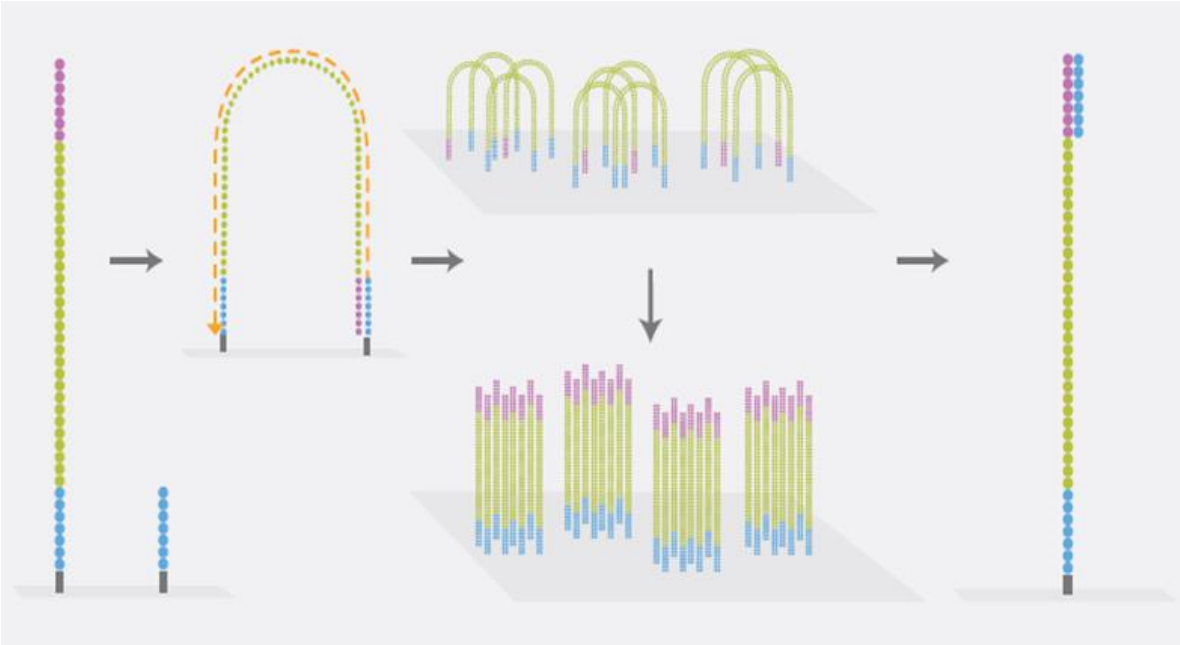


Figura 8. Amplificación clonal por el método del puente (Bridge-PCR). La PCR ocurre en los cristales llamados Flow cell en los cuales tiene lugar la secuenciación. Los fragmentos de ADN se unen al adaptador que también actuará como cebador. Cada fragmento de ADN forma un clúster de fragmentos de ADN idénticos. **https://www.cegat.de/**

El proceso para la secuenciación en cada colonia se hace de la siguiente manera[23]:

1. Aportas a la placa una mezcla con todos los nucleótidos marcados, con un fluorocromo que indica un color para cada base. El propio fluorocromo, cuando se incorpora, impide la polimerización.
2. Se lava para quitar los nucleótidos no incorporados.
3. Se captura la fluorescencia en una imagen. Esta imagen corresponde a un ciclo. En cada ciclo, para cada colonia (que representa una lectura) sale un color cuyo color se corresponde con el del nucleótido añadido.
4. Los colores se convierten en letras (**conversión digital**). Tenemos veinte millones de puntitos decolores por cada *flowcell.*
5. Comienza un nuevo ciclo y la máquina elimina el fluorocromo del último nucleótido añadido, dejando un extremo hidroxilo libre y permitiendo que continúe la polimerización.
6. El proceso de repite por un número de veces igual a longitud de la secuencia que vamos a obtener, dando lugar a un número de ciclos idénticos a la longitud de la secuencia de los fragmentos.

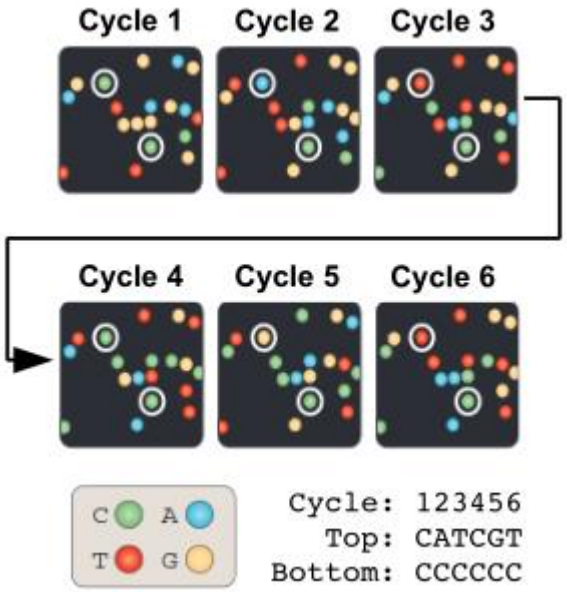


Figura 9. Imágenes de cada uno de los ciclos producidos en la secuenciación en puente. Puede verse como se va secuenciando simultáneamente las distintas colonias en cada uno de los ciclos. Adaptado por Luis del Peso de Nat. rev. Genetics (2010)

Este proceso está automatizado y nos devuelve como resultado una secuencia en formato fastq, en el que cada fragmento se representa en cuatro líneas, donde la primera comienza con un arroba y es información técnica del procedimiento, la segunda es la secuencia obtenida a partir del análisis de las imágenes (es la información biológica que nos interesa), la tercera es un espaciador (+) y la cuarta presenta una secuencia de caracteres que codifican la probabilidad de que la base se haya llamado correctamente.

- **Control de calidad de las lecturas** (eliminación de lecturas de baja calidad) → Al final podremos obtener reads de alta calidad a partir de los fastq producidos en la secuenciación. Se utilizan programas como FastQC[24] para realizar este análisis.
- **Alineamiento a un genoma** (o transcriptoma) de referencia o ensamblarlos en *contigs* para hacer un ensamblaje *de novo*. Hay una gran cantidad de programas que se dedican a realizar esta actividad. La clave de estos programas es ser lo suficientemente rápidos para que el alinear millones de secuencias cortas sea factible en un marco de tiempo aceptable, pero que tenga una flexibilidad suficiente para saber en qué zonas puede alinearse y asignar correctamente aquellas lecturas que mapean en localizaciones múltiples; a la vez que pueda tratar los problemas de la eliminación de intrones en el ARNm eucariótico. Los programas van mejorando en estos factores, y los desarrollos tecnológicos que permiten secuenciar lecturas cada vez más largas pueden disminuir la probabilidad de que las lecturas puedan unirse en distintos puntos.
- **Cuantificación:** El objetivo es obtener, para cada transcrito descrito en un genoma completo, la cantidad de lecturas que alinean en él. La cuantificación a nivel de transcrito depende de modelos probabilísticos para estimar la abundancia de cada isoforma, ya que muchas de ellas comparten ciertas regiones exónicas y es difícil atribuir a que isoforma pertenece cada una de

ellas. Algunos métodos cuantifican a nivel de gen, olvidándose de estas isoformas y colapsando todas las lecturas en el gen concreto.

- **Expresión diferencial**: Una vez hemos obtenido esta información, podemos comparar la expresión entre diferentes condiciones (2.1.3 Expresión diferencial)
- **Validación:** Los experimentos de transcriptómica pueden validarse utilizándose técnicas de qPCR.

Es importante tener un buen manejo y conocimiento de Linux, ya que la mayoría de los programas a utilizar deben lanzarse a través de la línea de comandos ya que carecen de interfaz gráfica. Suele ser necesario tener acceso a gran cantidad de memoria RAM para el procesamiento de toda la información, así como espacio suficiente en el disco para tratar con archivos de grandes dimensiones. El conocimiento de uno o varios lenguajes de programación ayudará en todo el proceso, ya que facilitará la automatización, generalización, comprensión y capacidad de paralelizar todos estos procesos, permitiendo incluso generar programas que realicen varias funciones de manera simultánea al utilizar de forma interna uno o varios de los ya conocidos. Se necesitan realizar distintos *scripts* que ayuden a adaptar las salidas de un programa para la entrada del siguiente y estandarizar las mejores condiciones en las que se realice este proceso.

2.1.2.3 Principales problemas de los estudios transcriptómicos

Existen importantes sesgos respecto a la expresión de determinados transcritos: En el RNA-seq se favorece mucho los genes que se expresan más, por lo que las isoformas o genes minoritarios son difíciles de detectar en muchas ocasiones, y en estos podría radicar la diferencia que queremos observar. Estos genes que tienen una alta tasa de transcripción presentan una mayor cantidad de transcritos, incrementando las probabilidades de ser secuenciado, dejando a los genes de baja transcripción con una cobertura mucho menor en perspectiva.

El análisis de los datos de RNA-seq no está tan estandarizado y no existe un consenso claro para definir el preprocesado, normalización y qué métodos son mejores para realizar la inferencia [13]. Además, se sabe que la expresión génica es mucho más variable que la propia secuencia del ADN y tanto a nivel de individuo como de poblaciones y especies, incluso entre condiciones. En un mismo individuo, se pueden encontrar diferencias en el mismo tejido, entre diferentes células de este.

2.1.3 Expresión diferencial

Además de conocer lo que se está expresando en un determinado momento y lugar, se puede tener en cuenta otra variable adicional: no todos los genes que están expresándose sufren el mismo procesamiento. Un mismo ADN expresado puede dar lugar a distintos ARN maduros (isoformas) en función de los intrones y exones que incorpore, el sitio de inicio y final de la de la transcripción… Por este motivo, para el estudio de la expresión tenemos dos opciones: colapsar todas las lecturas y mirar la expresión a nivel de gen (más sencillo, aunque menos exacto e informativo) o intentar determinar de qué isoforma viene cada una de las lecturas y hacer una reconstrucción de estas (mucho más complejo, y un paso computacional adicional). Es importante que distingamos bien entre la expresión génica (qué genes están activos transcripcionalmente en un determinado momento), la expresión diferencial de exones (DTE) y uso diferencial de transcritos (DTU). El primer caso considera los niveles de los transcritos de manera individual mientras que el segundo son las proporciones de las isoformas del gen[25]. Como

se muestra en la figura 10, el DTU siempre implica un DTE, pero no tiene por qué ocurrir en caso contrario.

- **DTE:** Se utiliza la cuantificación normalizada de las lecturas (TPM/CPM) como medida de la expresión de los transcritos y se utilizan modelos de análisis de expresión diferencial génica, como, por ejemplo, limma.

- **DTU**: Normalizamos la expresión de cada transcrito respecto al total de la suma de los transcritos. Θ_k= Expresión relativa de un transcrito k; k=1, …, K respecto al set total de transcritos, con $\Theta_k \geq 0$ y $\sum_1^K \Theta_k = 1$

  Suele utilizarse para detectar cambios en las proporciones de isoformas o *switching* de isoformas. La expresión global del gen no tiene por qué haber cambiado, y que, con la misma cantidad de ARN produciéndose, toda su maduración lo haga derivar hacia una nueva isoforma que estaba en mucho menor proporción y cuya funcionalidad puede no tener nada que ver con la que estaba en una mayor proporción en el caso anterior.

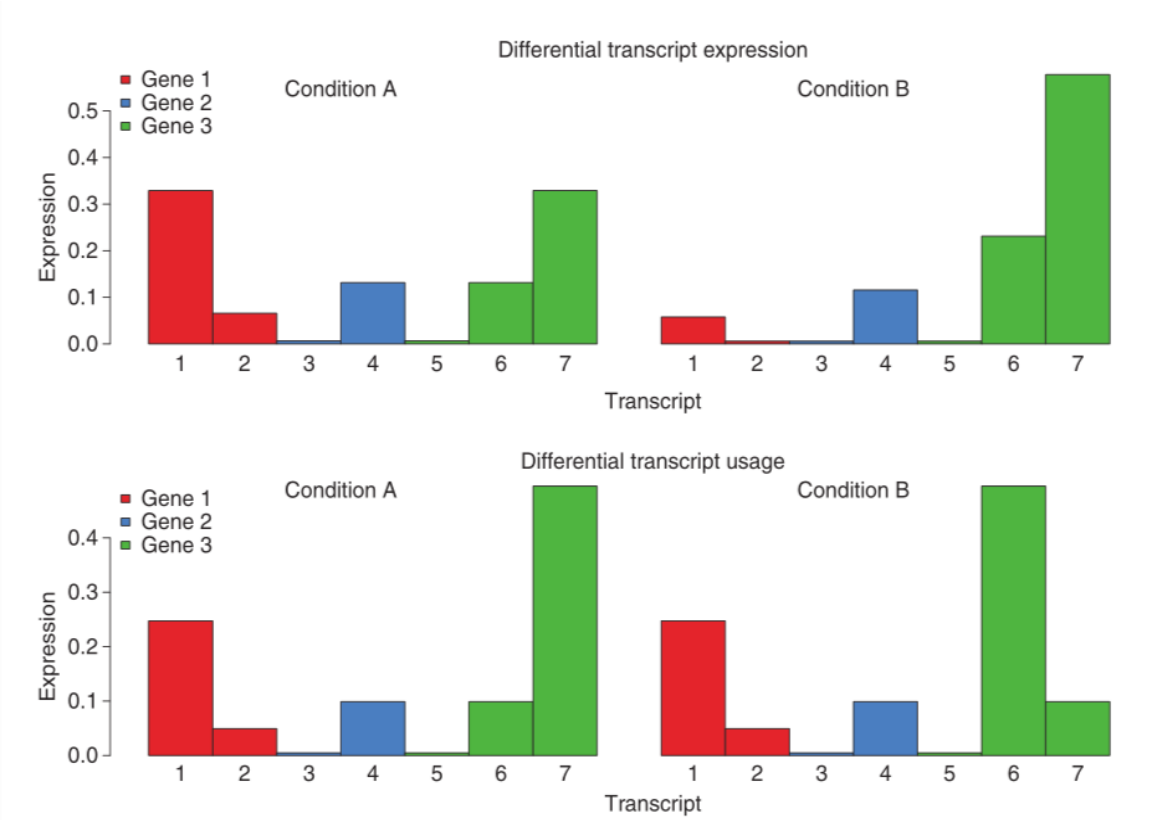


Figura 10. Expresión diferencial de transcrito (arriba) y uso diferencial de transcrito (abajo). Vemos como en el caso del uso diferencial de transcrito las proporciones totales se mantienen para cada gen tanto si hay diferencias en el uso como si no. Papastamoulis, P. & Rattray, M. Bayesian estimation of differential transcript usage from RNA-seq data. Statistical Applications in Genetics and Molecular Biology 16, 387–405 (2017).

2.1.4 Splicing alternativo

El proceso de splicing del ARN o empalme del ARN es un proceso por el cual, en ARN recién transcrito en el núcleo, a la vez que se producen otros procesos de maduración (adición de la caperuza (cap) en el 5' y la cola de poliadeninas en el extremo 3'), sufre cortes y empalmes, eliminando los intrones no codificantes para crear una secuencia continua de exones codificantes. Una vez maduro, es exportado al citoplasma. Este proceso es mediado por el espliceosoma, un complejo de varias proteínas que se unen al ARN. Hay cuatro regiones que participan en el reconocimiento, muy conservadas, los motivos de reconocimiento de splicing. Esta gran conservación es señal de su gran importancia biológica. Estas regiones son: i) El sitio donante, el GU del 5' del intrón; ii) el sitio aceptor, el AG del 3' del intrón; iii) el punto de ramificación, que presenta una adenina (A), que se encuentra cerca del 3' del intrón; iv) Tracto de polipirimidinas de 15 nucleótidos (PPT) que se encuentra entre el 3'AG y el punto de ramificación[26]. Estas zonas son clave para el reconocimiento por un complejo proteico formado por ribonucleoproteínas nucleares pequeñas (snRNP), el espliceosoma. Estas snRNP están formadas por 5 ARN nuclear pequeño (snARN), U1, U2, U4 y U5 que son útiles en el reconocimiento, y por 7 proteínas Sm comunes (B, D1, D2, D3, E, F, G) y proteínas Sm específicas del snRNP.

Lo que tiene que producirse es un corte en una frontera y la generación de un nuevo enlace fosfodiéster en el otro extremo. El proceso comienza con el reconocimiento de la frontera exón-intrón por U1, el 5' GU. Posteriormente, U2 reconoce el punto de ramificación y aparea con él, a excepción de la adenina sombreada en rojo, que será el nucleófilo de la reacción de transesterificación (Figura 11**a**). U5, U4 y U6 trimerizan, y U5 se une al exón 5' y U6 a U2 formando un complejo inactivo. Posteriormente, se desplaza U1, U6 uniéndose al extremo 5' AG y a U2 (Figura 11**b**). U6 y U5 catalizan la reacción de transesterificación entre el punto de ramificación (nucleófilo) y el 5' GU, que queda con un extremo OH libre. Posteriormente, y gracias a que U6 actúa como puente, este OH libre puede actuar como nucleófilo sobre el extremo 3' del intrón, liberándolo por completo (Figura 11**c**).

El splicing alternativo o empalme alternativo es un proceso por el cual pueden generarse diferentes versiones de transcritos maduros a partir de un único gen (Figura 12**a**), en función de las regiones que permanezcan o se eliminen en el producto final, mediante la retención de intrones, exclusión de exones, comienzos de transcripción alternativos… (Figura 12**b**).

Este proceso de splicing o ayuste alternativo es posible gracias a diferentes elementos reguladores que favorecen el reconocimiento de ciertas zonas y ocultan otras en determinadas circunstancias. Existen algunos elementos en *cis* conocidos como Elementos reguladores de splicing o SREs que ayudan a definir la especificidad del splicing, actuando como silenciadores o y pudiendo existir tanto en intrones como exones[27]. Los SREs reclutan diferentes factores de splicing en trans, favoreciendo o inhibiendo el uso de un sitio de splicing adyacente, y controlando de esta manera el splicing.

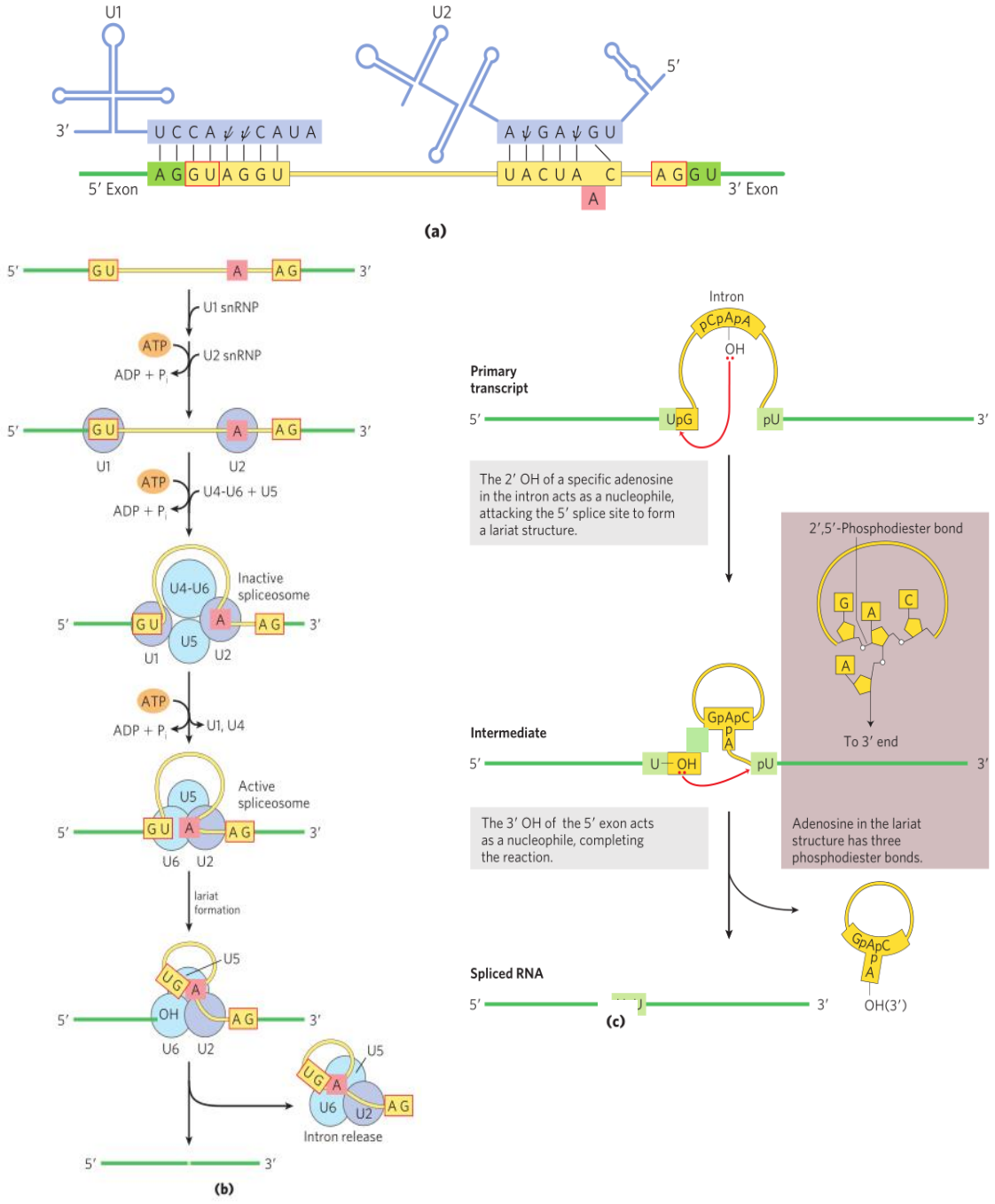


Figura 11. Mecanismo de splicing alternativo. El espliceosoma se une a un pre-ARNm, gracias a las regiones conservadas permitiendo que se produzca el proceso de corte y empalme. **A**) Interacción de las snRNP con el ARN. **B**) Ensamblaje del espliceosoma. Se unen los complejos U1 y U2 y posteriormente el complejo U4, U5 y U6 se añaden para formar un espliceosoma inactiva, que tras unas reorganizaciones y la liberación de U1 y U4, permite activar el espliceosoma y favorecer la reacción. **C**) Reacción de transesterificación donde el OH de la Adenina del punto de ramificación actúa como nucleófilo, atacando al 5' GU, dejando el U-OH libre para realizar un segundo ataque nucleófilo, esta vez sobre el extremo 3' AG

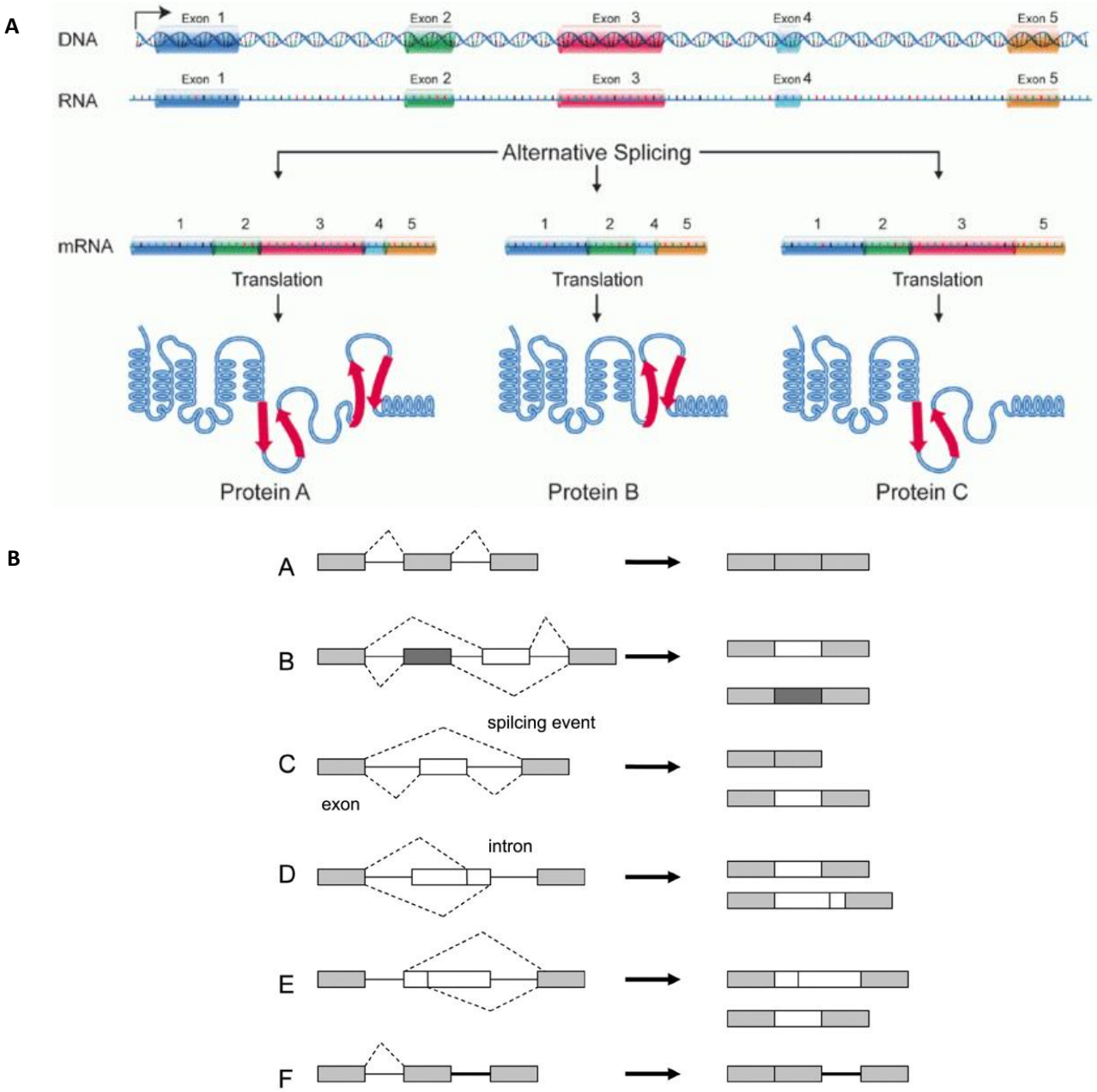


Figura 12 . Mecanismos de splicing alternativo. A) Podemos ver como a partir de un único ARNm sin procesar se generan distintos ARNm maduros, cada uno incluyendo distintos exones. Algunos de ellos se traducen y dan lugar a proteínas, eliminando el concepto de un gen, una proteína. B) Diferentes mecanismos de producción de splicing alternativo. El A representa el splicing constitutivo, con la eliminación de todos los exones; en el B se muestra exones mutuamente excluyentes, en el C un *cassete* del exón, exitrones; en el D, un sitio alternativo de sitio aceptor (3'); en el E, un sitio donante 5' alternativo y en el F retención de intrón.

Por ejemplo, el procesamiento diferencial de la tropomiosina lleva en estudio desde hace muchos años. Se sabía que existía dos exones que eran mutuamente excluyentes (exón 2 y 3), y que esto se debía a la proximidad del punto de ramificación del exón 3 al sitio donante del 5' del exón 2, que se presenta más cercano al 5' en vez de al 3', como viene siendo habitual (Figura 13). Esta cercanía evita, por impedimento estérico, la formación de un complejo de espliceosoma activo, impidiendo que los exones 2 y 3 salgan juntos. Además, se sabe que por defecto se incorpora preferentemente el exón 3 en vez del 2, con excepción de las células del músculo liso[28]. Esta preferencia puede deberse a una competición cinética de los factores de splicing por un determinado PTB o punto de ramificación concreto. Se ha visto que variaciones en el factor constitutivo de splicing ASF/SF2 pueden modular la selección de un 5' de splicing u otro. La elección del exón 3 sobre el 2 puede deberse al punto de ramificación del exón 3, que es muy similar al consenso. Además, presenta un tracto de polipirimidinas muy rico en este tipo de bases, haciéndolo más fuerte respecto al del exón 2.

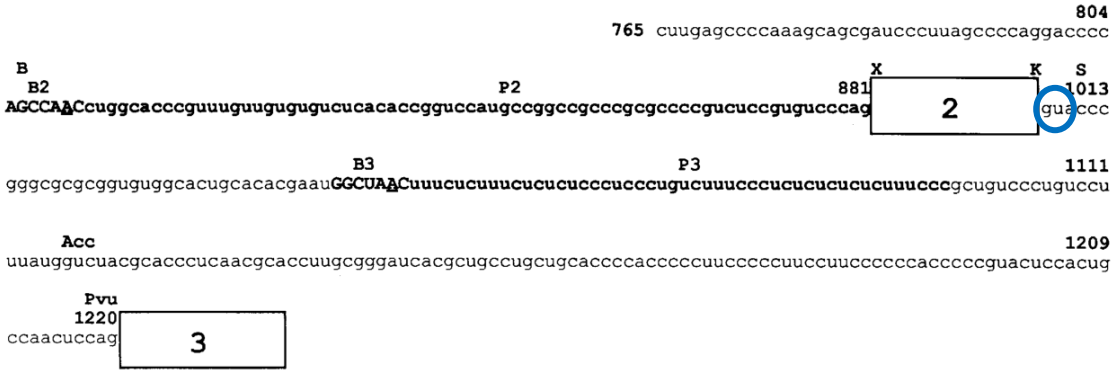


Figura 13. Secuencia nucleotídica de los intrones entre los exones 2 y 3 de la α-tropomiosina. Los exones se muestran en cajas con su correspondiente número de exón. P2 y P3 representan los tractos de polipirimidinas, destacados con minúsculas y negrita, de los exones 2 y 3 respectivamente. B2 y B3 representan los puntos de ramificación de los exones 2 y 3, respectivamente, remarcados en negrita y mayúscula. La A responsable del ataque nucleófilo se representa subrayada. Rodeado en azul se marca el 5' GU donante del exón 2, muy cercano al B3, impidiendo que puedan incorporarse ambos exones simultáneamente.

Respecto a cómo se produce la selección alternativa de uno u otro exón en función del tejido, se sabe que el balance de los diferentes componentes que forman el espliceosoma y la afinidad de este a distintas regiones de los ARNm puede favorecer un tipo de regulación y otra. Existen tres isoformas relevantes en splicing de la proteína de unión a tracto de pirimidinas (PTB): PTB1, PTB2 y PTB4. PTB produce una suave represión en el exón 3 de la α-tropomiosina, pero en el músculo liso, esta represión es aún mayor: esto se debe al uso de las diferentes isoformas; la PTB1 reduce el salto del exón 3 de la tropomiosina mientras que PTB4 produce un aumento en el salto del exón 3. En el músculo esquelético, se produce la eliminación del exón 3 permitiendo la permanencia del exón 2, ya que son mutuamente excluyentes (Figura 14)[29]. Se ha descrito este incremento de la represión de las isoformas, graduado de la siguiente manera: PTB4 > PTB2 > PTB1.

A pesar de que hace años que se ha refutado la teoría de un gen-una proteína, el origen de la variabilidad biológica todavía está en estudio. Teniendo en cuenta que el splicing alternativo es una gran fuente de diversidad, podría considerarse como una de las causas de esta. Sin embargo, hay bastantes controversia sobre la importancia biológica de las distintas isoformas y sobre si muchas de ellas llegan tan siquiera a producir proteínas[30], [31]. Esta dualidad se repite constantemente, ya que mientras que en algunos

de los artículos se describe como al menos el 75-85% de las isoformas que se encuentran en abundancia media-alta se han detectado en los unidas a los ribosomas[32], otros comentan que, a pesar de la gran cantidad de ARN mensajeros que llegan a poderse identificar durante un experimento de RNA-seq, solo un 1% genoma humano (240 genes) muestran evidencia de que hay más de una isoforma proteica relevante[33], [34]. Además, se realizaron estudios de los productos de splicing anotados en ENCODE y se vio que la mayoría de las isoformas producirían estructuras 3D muy alteradas y que por ende tendrían cambios dramáticos en la función si se tradujesen a proteína[35].

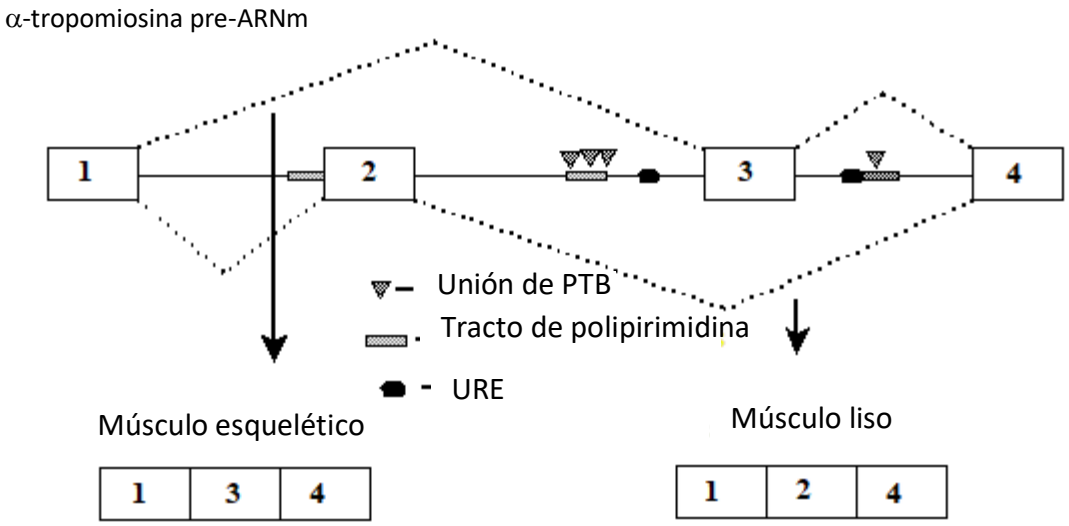


Figura 14. Splicing alternativo de la tropomiosina. En función del tejido en el que se encuentre, se expresarán una isoforma u otra de la proteína PTB, favoreciendo un determinado tipo de splicing y la inclusión excluyente entre los exones 2 y 3, debido a la cercanía entre el sitio de ramificación del exón 3 y el sitio donante 5' del exón 2. URE: Elemento regulador anterior

Las pocas evidencias de isoformas proteicas detectadas suele asociarse a las limitaciones técnicas de la espectrometría de masas, que ya incluso en proteínas altamente expresadas no cubre la totalidad de su expresión, por lo cual las isoformas no canónicas que se encontrasen a bajos niveles, aunque se tradujeran, no se detectarían[36]. En la misma línea de búsqueda de la diversidad del ARN en la proteómica, se analizaron péptidos de varias bases de datos, encontrando solo 282 isoformas en los 12,716 genes (2.22%), siendo la mayoría de ellas además simplemente pequeñas modificaciones de la variante principal, generalmente por sustitución de exones homólogos[37]; sin embargo se sabe que estas modificaciones sí que son frecuentes y tienen gran relevancia en ciertos tipos celulares, especialmente en corazón y cerebro[37].

La mayor parte de las isoformas de ARN anotadas suelen producirse por ganancia o pérdida de exones[38]. De las pocas isoformas proteicas cristalizadas en el PDB, la mayoría provienen de sustituciones homólogas de exones alternativos (*cassete* de exones), que normalmente no causan un cambio en la pauta de lectura y producen una proteína bastante similar, y pocos de los splicing producían rupturas en los dominios de Pfam[37]. Sin embargo se han descrito diferentes mapas de interacciones proteína-proteína o proteína-ADN entre las isoformas proteicas y las de referencia, por lo que esta diversidad sí se muestra en algunos aspectos[39]. Además, se han descrito algunas isoformas proteicas

en humanos que presentan funciones antagónicas en función del procesamiento que hayan sufrido como en el caso del cáncer y HRAS o VEGFA (Figura 15)[40] y como el desbalance en la expresión de un factor de splicing puede afectar al desarrollo neuronal[2]. En otros casos, se ha visto como el splicing alternativo permite la eliminación de determinados dominios de unión de fármacos haciendo a las células resistentes según el procesamiento, como en el caso de las isoformas de BRAF[41]

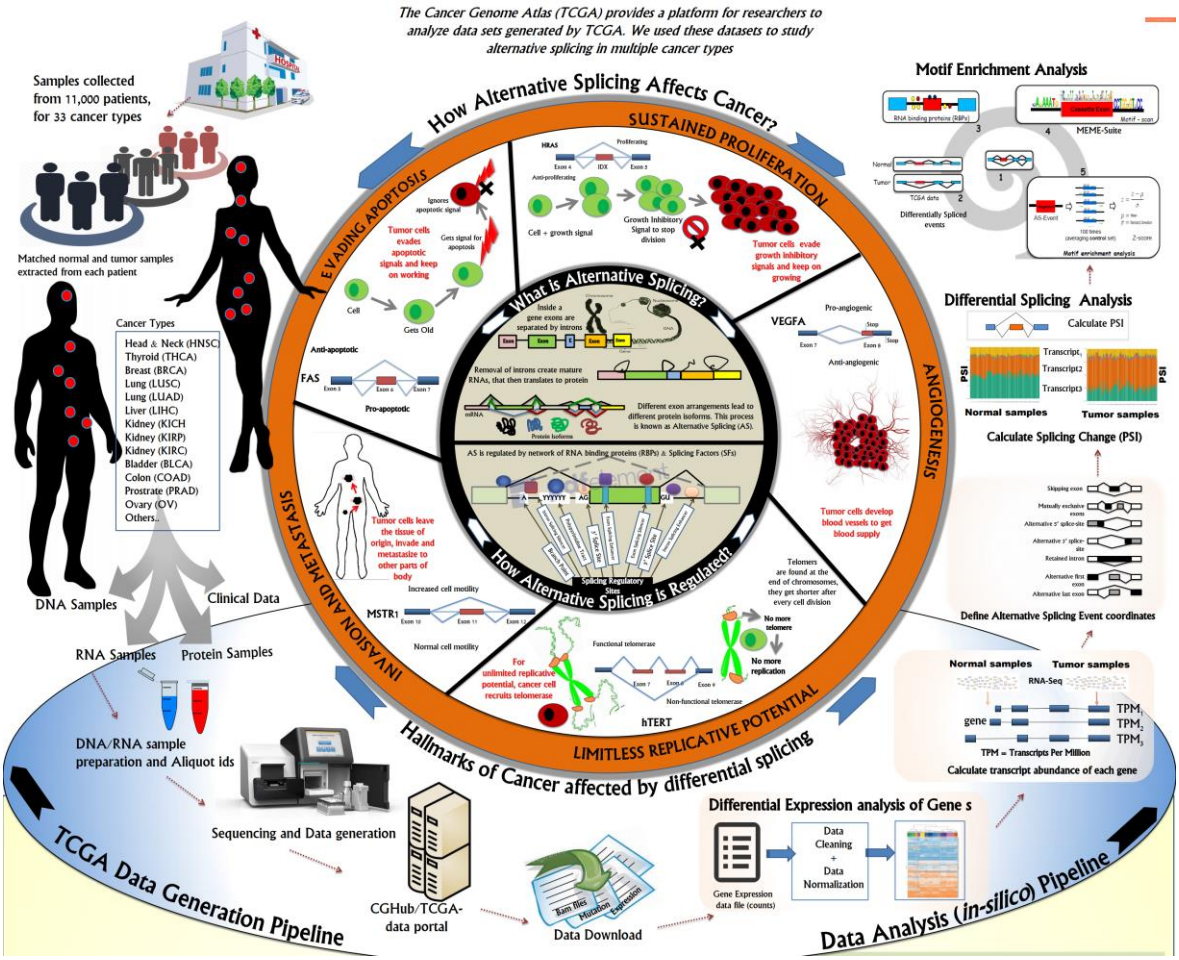


Figura 15. Splicing alternativo como conductor del cáncer. Singh, B. et al. (2016) 4th DCEXS Symposium.

Uno de los principales problemas es que no existe un procedimiento estandarizado por el que se pueda estudiar de forma rutinaria en el laboratorio el *splicing* alternativo, por lo que no se suele incorporar en los estudios generales de expresión, cuando sin embargo se ha demostrado que aporta una capa de información adicional (Figura 16)[42]. Además, se ha descrito como la existencia de un cambio en la proporción de isoformas (*isoform switching*) puede provocar el incremento falsos positivos en estudios de expresión génica[43]. Por tanto, urge establecer un protocolo que se pueda seguir en estos casos, especialmente de manera sencilla para que pueda convertirse en una práctica rutinaria menos laboriosa. Actualmente, existen una gran cantidad de programas distintos para estudiar el splicing alternativo, basados en algoritmos distintos y aproximaciones muy diferentes. Existen pocos estudios que comparen las distintas herramientas y sus comportamientos en distintas condiciones, especialmente las más frecuentes en la práctica por motivos de protocolo o económicos. Por ese motivo, nuestro objetivo fue

comparar las distintas herramientas existentes en el estudio del análisis de splicing alternativo y analizar las eficiencias de los mismo.

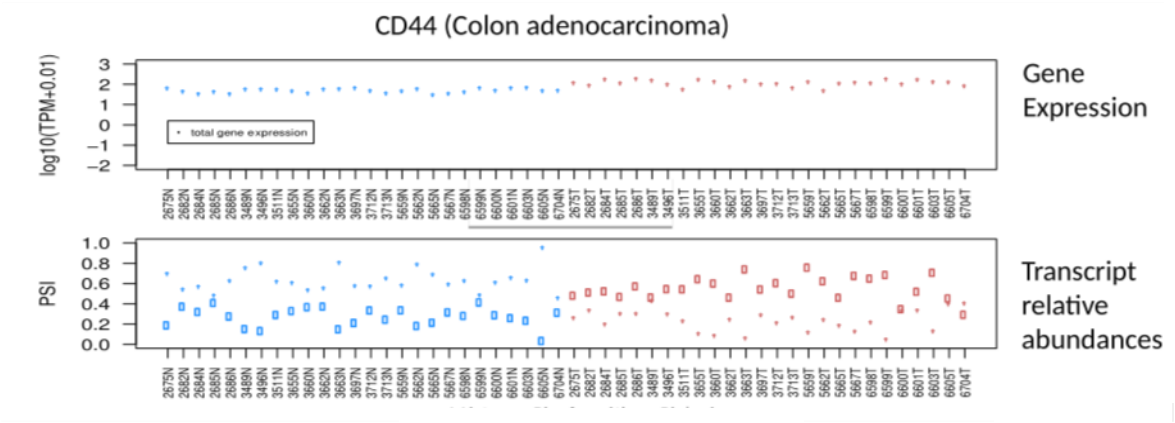


Figura 16. Detección de switching de splicing alternativo en muestras de tumor que definen firmas de expresión en el cáncer. *Nucleic Acids Res.* Sebestyén, E., Zawisza, M. & Eyras, E. 43, 1345–1356 (2015).

2.2 Generación de lecturas

2.2.1 Polyester

Con el objeto de estudiar la diferente eficiencia y las condiciones óptimas para el estudio del splicing alternativo, necesitamos obtener lecturas, equivalentes a los resultados de secuenciación que se reciben de *Illumina*, pero simuladas, intentando reproducir en lo posible la variabilidad biológica y experimental.

Polyester es una de las varias herramientas que permiten generar lecturas para poder testar controlando diversos parámetros experimentales. Está disponible como un paquete de R y permite simular estas lecturas con diferente señal de expresión diferencial (un mayor o menor cambio en el incremento o *fold change*), situándolo con una mayor ventaja respecto a otros programas como FluxSimulator o Beers, que no tienen como opción propia este parámetro. Otros simuladores como RSEM requieren un paso adicional de alineamiento de lecturas reales, lo cual haría más lento el proceso[8]. Además, en nuestro caso, este último será parte de los distintos pipelines a analizar, por lo que evitar la simulación por el mismo evita la aparición de sesgos.

Otra característica interesante de Polyester y uno de los motivos adicionales de su elección es que permite implementar modificaciones en el sesgo de GCs, cuyo parámetro es bastante interesante y problemático en los estudios de transcriptómica. Este sesgo se define por la dependencia existente entre la proporción de GCs en una región concreta del genoma y el número de lecturas que mapean en ella. Además, debido a que la proporción de GCs es heterogénea por todo el genoma y que tiene cierta relación con la funcionalidad, es muy difícil de distinguir de la señal verdadera. Su efecto que no se reduce con los replicados, pues no es consistente entre experimentos ni librerías del mismo experimento, por lo que es un campo de estudio muy importante y determinar que pipeline es capaz de evitarlo de mejor manera podría ser una manera de mejorar los procedimientos establecidos[9].

Para hacer la simulación, muchos de los parámetros de Polyester requieren analizar datos reales de RNA-seq para hacerlo lo más parecido posible a las lecturas que se obtendrían en la realidad. Para la obtención de esta información, los autores analizaron las lecturas de RNA-seq de siete replicados

biológicos que se encuentran en la base de datos pública GEUVADIS (*Genetic European Variation in health and Disease*), escogidos aleatoriamente entre las muestras de cada uno de los siete laboratorios implicados en la secuenciación de este estudio, que incluían muestras de 7 pacientes de tres poblaciones diferentes de HapMap: CEU (Residentes en Utah con ancestros del norte y oeste de Europa), TSI (Toscanos viviendo en Italia) y YRI (Yoruba viviendo en Ibadan, Nigeria). Estos datos se alinearon utilizando Tophat, ensamblado con *cufflinks* y procesados con el paquete de R Ballgown.

Una de las ventajas que presenta Polyester es la capacidad de determinar exactamente el número de lecturas por transcrito, de manera independiente para cada replicado en cada experimento. Además, para obtener la expresión en función de estas lecturas se puede elegir el uso de un modelo predefinido o aportar uno. El modelo asume que el número de lecturas para simular de cada transcrito se coge al azar de una distribución binomial negativa, que se sabe que es capaz de capturar la variabilidad biológica y técnica[44]. La distribución binomial negativa se adapta a la distribución de datos de secuenciación porque los resultados de secuenciación nos dan la cantidad de lecturas hay en una determinada región (gene o exón), es decir, datos cuantitativos discretos. Por gen, este valor tendrá una magnitud de entre una decena a miles de lecturas (en función de la expresión, pero también de la longitud del gen), mientras que las lecturas totales son del orden de decenas de millones, incluso más, dependiendo de la profundidad de la secuenciación: en consecuencia, la probabilidad de que una lectura mapee en una localización concreta es muy pequeña. Debido a esto, siempre se había utilizado una distribución de Poisson, procesos con baja probabilidad dentro un espacio muestral muy grande, ya que esta distribución representa la probabilidad de que ocurra un determinado número de eventos en cierto periodo de tiempo cuando estos tienen una frecuencia de ocurrencia media. Sin embargo, la variabilidad den el número de lecturas entre réplicas es mucho mayor de lo que la distribución de Poisson es capaz de modelar, ya que considera que la varianza de la misma es igual que la media (Figura 17) [45]. La distribución Binomial Negativa es muy similar, ya que representa el número de veces hasta que debe producirse un evento hasta que alcanzamos el éxito un número r veces, siendo un parámetro adicional para modelar dispersión. La explicación biológica radica que, al comparar diferentes condiciones, es común (y muy recomendable) la presencia de varios replicados para poder hacer una inferencia estadística. Estos replicados biológicos deben ser independientes, y suelen presentar cierta variación en cada muestra, incluso perteneciendo a la misma condición, produciendo esta "sobredispersión", observable en los datos de secuenciación. Este parámetro de varianza suele ser fácil de estimar en los experimentos de RNA-seq, ya que la mayoría de las herramientas nos permiten obtenerla para la densidad de lecturas que tenemos. Después, este parámetro se utilizará para modelar la distribución Binomial Negativa de cada gen.

$$\sigma^2 = \mu + \alpha\mu^2$$

Ecuación 1. Varianza de la distribución binomial negativa. La varianza en la distribución binomial negativa es función de la media, pero también de un parámetro de dispersión, alfa. Si este parámetro fuese cero, la varianza y la media serían iguales, cumpliéndose las condiciones de la distribución de Poisson y siendo por tanto esta válida. Alfa puede definirse como 1/r, siendo r el número de "éxitos" que queremos alcanzar.

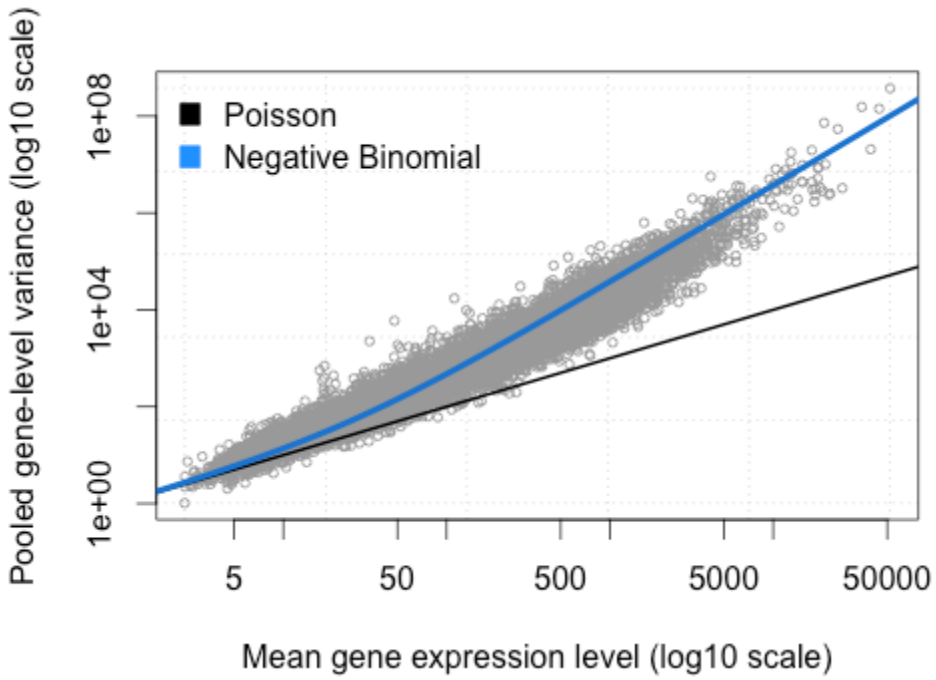


Figura 17. Distribución de la expresión media de los genes respecto a la varianza del experimento. De forma simultánea, se representa una distribución de Poisson y una binomial negativa, observando que la segunda se acerca mucho más a las lecturas respecto a la varianza (eje y).
Fuente: Lipp, J. Why sequencing data is modeled as negative binomial. (2016). Available at: https://bioramble.wordpress.com/2016/01/30/why-sequencing-data-is-modeled-as-negative-binomial/

En *Polyester* se sigue este modelo, definiendo la binomial negativa como

$$\mathbf{Y_{ijk} \sim Negative\ Binomial\ (media=\mu_{jk};\ tamaño=\ r_{jk});}$$

Siendo Y_{ijk} el número de lecturas simuladas para el replicado i, en la condición experimental j y para el transcrito k. Se puede proporcionar μ_{jk} para cada transcrito k y condición experimental j. Además, si reducimos el parámetro alfa a cero ($r_{jk} \to \infty$), podemos tener una distribución de Poisson que nos valdría para analizar replicados técnicos si los tuviésemos en alguna ocasión.

La mayor facilidad que presenta Polyester es la posibilidad de definir la expresión diferencial por medio de una proporción de cambio (*fold change*) cuando solo se analizan dos condiciones. El proceso consiste en asignar la misma media a ambas condiciones y después multiplicar esta media por el valor de *fold change* dado para cada transcrito. Dispone además de una gran cantidad de opciones de personalización para el contaje de lecturas, permitiendo complicar los modelos tanto como se desee.

Una vez se han especificado todas las condiciones, Polyester simula el experimento de RNA-seq, empezando por el paso de fragmentación, pudiendo tomar el tamaño de una distribución normal o de una distribución biológica, estimada de los alineamientos previamente mencionados de GEUVADIS. Posteriormente, los fragmentos pueden estar distribuidos uniformemente entre los transcritos o introducir un sesgo, ya que se sabe que la cobertura no es uniforme para todos los transcritos, dando la opción de escoger entre dos tipos de sesgos de diferentes protocolos de fragmentación.

A la fragmentación, en un experimento biológico le sucede la secuenciación de estos fragmentos. Para ello, *Polyester* simula un protocolo *unstranded* (donde no sabemos la dirección de la cadena de la que proviene cada una de las lecturas) como la mayoría de los protocolos de *Illumina*. En este sentido, cada fragmento generado desde el genoma/transcriptoma proporcionado puede generar la reserva-complementaria con una probabilidad de 0.5.

2.3 Alineamiento de las lecturas

Una vez hemos obtenido las lecturas secuenciadas en formato fasta o fastq, necesitamos producir el alineamiento de cada una de estas lecturas a la referencia, de la misma manera que ocurre en los experimentos de expresión génica, para poder observar la cantidad de expresión de cada uno de los transcritos. En RNA-seq, el alineamiento de secuencias presenta de las lecturas genómicas. Esta complicación radica en que en los experimentos de RNA-seq se producen millones de lecturas (secuencias cortas de nucleótidos) que debemos alinear frente a una referencia, que será generalmente el genoma. A diferencia de lo que ocurre con las lecturas del ADN, en el caso de el ARN este ya estará procesado y habremos perdido parte del mismo (los intrones) por lo que alinearemos fragmentos cortados y empalmados contra su secuencia original, teniendo *gaps* o huecos con material genético inexistente que pertenecerán al intrón/es que se han eliminado[46]**.** Es decir, mientras que en el genoma de referencia tendríamos una región intrónica, nuestra lectura podría ser la interfase entre dos exones que han eliminado esa misma. Debido a esto, a pesar de que se han desarrollado una gran cantidad de métodos computacionales para que las lecturas puedan alinearse al ADN de referencia, ninguno de ellos todavía muestra una eficacia cercana a la de sus equivalentes en genómica. Además de esta problemática, se suma el hecho de que a pesar de ser fragmentos cortos tienen una importante tasa de error.

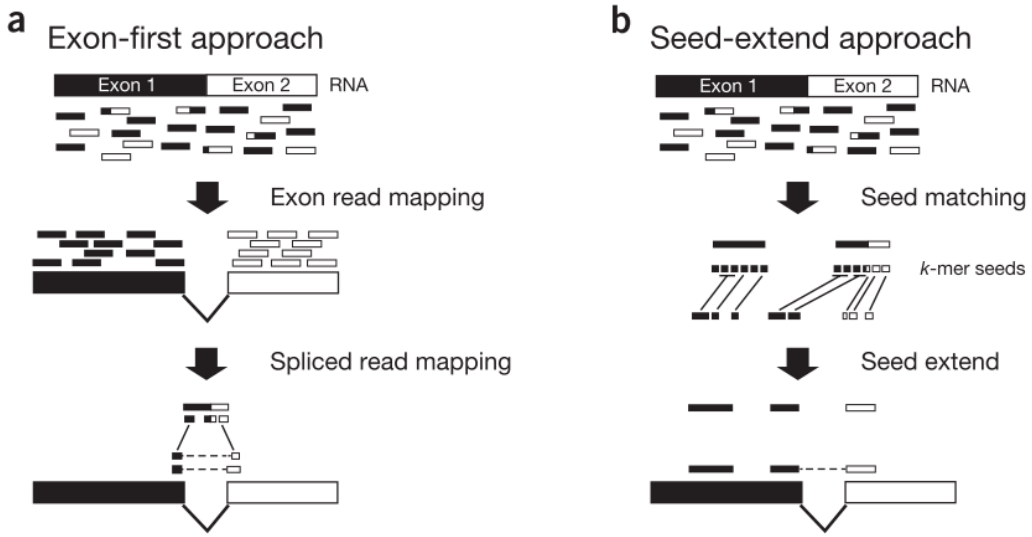


Figura 18. Estrategias para alinear las lecturas de ARN frente al genoma de referencia. (a) Las lecturas que pueden alinearse perfectamente se alinean. Las que no se pueden alinearse sin una gran cantidad de huecos o errores se rompen en trozos más pequeños y se prueban a alinear de nuevo. (b) Desde el principio rompemos las lecturas en trozos más pequeños para evitar que en una misma lectura caigan regiones de dos exones contiguos que en el genoma de referencia tengan una región intrónica en medio.
Garber, M., Grabherr, M. G., Guttman, M. & Trapnell, C. Computational methods for transcriptome annotation and quantification using RNA-seq. *Nat. Methods* 8, 469–477 (2011).

2.3.1 STAR

Es un algoritmo de alineamiento cuyo nombre proviene de "Alineamiento de transcritos que han sufrido splicing a una referencia", del inglés '*Spliced Transcripts Alignment to a Reference* (STAR)'. Fue publicado en 2013 y supuso una gran mejora respecto al método actual más extendido (Tophat [2009] que era una adaptación del algoritmo Bowtie utilizado en genómica, pero adaptándolo para la presencia de grandes deleciones producidas por la maduración por splicing). Además, es un alineador especialmente diseñado para poder lidiar con mayores longitudes de lecturas, en respuesta a los mecanismos de secuenciación de tercera generación. Se ha producido directamente para trabajar con experimentos de ARN, por lo que no es una adaptación de alineadores de lecturas contiguas, sino que directamente alinea secuencias no contiguas directamente al genoma de referencia, produciendo que alinee muy bien secuencias que presentan errores, inserciones y deleciones. Es un algoritmo que consiste en dos pasos[47]:

1. Búsqueda de las semillas: MMP

   Las siglas MMP provienen de Prefijo Máximo Mapeable, en inglés *Maximal Mappable Prefix*. Dada la secuencia (R) de una lectura concreta, la localización i de esta y un genoma de referencia G, el prefijo máximo mapeable (MMP) consiste en la subsecuencia más larga que presenta una o varias secuencias de lecturas del genoma G, siendo MML (*Maximal Mappable Length)* la longitud más grande encontrada. Es decir, dada una lectura concreta, el algoritmo busca la posición o posiciones del genoma donde alinea perfectamente con la subsecuencia más larga posible de su secuencia: en fragmentos que idealmente no caigan en regiones de splicing, esto podría ser toda la longitud de lectura, y en el caso de que esté formado por dos exones, este mecanismo permite identificarlo. Por ejemplo, en la Figura 19**Figura 19.** Representación de la búsqueda de MMPs realizada por STAR para buscar sitios de splicing (a), errores en el alineamiento (b) y adaptadores o colas (c). podemos observar una lectura que cae justamente en una región intermediaria, que, como explicamos, pertenecería a una de estas lecturas que no puede alinear con el ADN de referencia. Se produce una búsqueda secuencial para que se produzca un alineamiento en la longitud máxima posible de su secuencia en una o varias posiciones del genoma, en este caso, pudiendo alinear solo el fragmento previo a la región de splicing, el MMP1. Esto reduce drásticamente la búsqueda de todos los posibles lugares de mapeo exactos. El fragmento o semilla de la lectura que ha quedado sin alinear vuelve a realizar la búsqueda del prefijo máximo mapeable, es decir, en qué región puede alinearse manteniendo la máxima longitud posible; en este caso la secuencia alineará en el sitio aceptor del splicing. Si en el proceso de extensión no se obtiene ninguna localización adecuada de alineamiento permite identificar regiones de colas de poli A, adaptadores de secuencias o suelen ser regiones del final de la amplificación, cuya calidad es mucho peor. La búsqueda de MMP se produce en las dos direcciones de la cadena y puede seleccionarse un punto de inicio por el usuario, evitando regiones de gran densidad de error.

   Detectar una región de splicing con esta técnica es posible sin necesidad de utilizar ninguna base de datos anotada, y sin necesidad de utilizar pasos previos de alineamientos contiguos. El algoritmo se implementa mediante el uso de matrices de sufijos no comprimidas, lo que le permite alcanzar gran velocidad, pero requiere una gran capacidad de memoria. Adicionalmente, STAR permite también detectar errores o cambios en la secuencia (*missmatch*), así como inserciones y deleciones, como puede observarse en la Figura 19**b**. Si al

final de la búsqueda, algún MMP se ha quedado sin mapear con el genoma de referencia debido a la presencia de algunos cambios de base, los MMPs que ya han sido alineados se utilizarán como anclas que pueden extenderse, con una mayor permisividad para los errores. Esta técnica sería similar a las del grupo de extender la semilla que se comentaba en el apartado anterior, Figura 18**b**. Si incluso tras la extensión la calidad es muy mala, esta semilla se descarta por tener baja calidad o incluso debido a ser un adaptador.

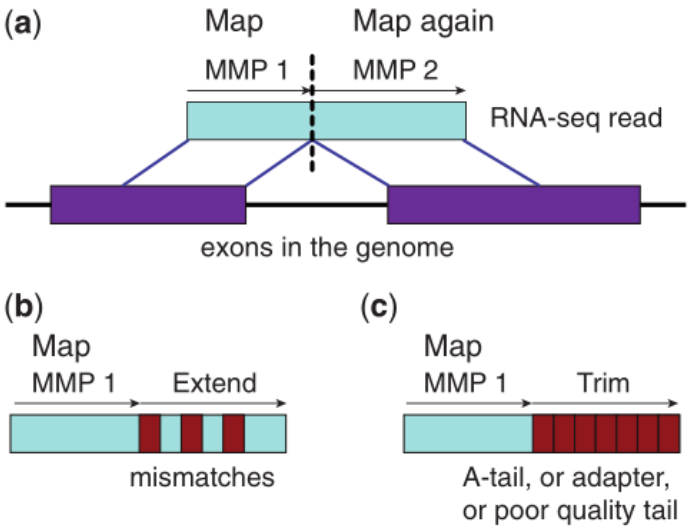


Figura 19. Representación de la búsqueda de MMPs realizada por STAR para buscar sitios de splicing (a), errores en el alineamiento (b) y adaptadores o colas (c).

2. Agrupación y puntuación: Construye un alineamiento mediante la unión de las semillas que alineó en primer lugar con el genoma.

    a. Agrupa las semillas por proximidad hacia unas semillas que actúan como ancla (serían el centro del clúster). Esta semilla suele definirse por los MMPs que solo presentan mapeo de su secuencia más larga en un único lugar. Todos los MMPs que alinean en una ventana de nucleótidos (definida por el usuario) alrededor de las anclas se unen asumiendo un modelo local y lineal. Para la unión de fragmentos se utiliza un modelo de programación dinámica que permite cualquier número posible de errores, pero solo una inserción o deleción (gap). El tamaño de esta ventana determinará el tamaño máximo del intrón definido, que variará según la especie, ya que representa la máxima distancia que pueden presentar en el genoma de referencia dos lecturas que deberían mapear juntas. Si las lecturas son *paired-end* se agrupan y unen conjuntamente, permitiendo un gap entre ellas o un solapamiento. Considera todas las combinaciones colineales posibles para cada clúster y escoge la que tenga una mayor puntuación.

Posteriormente, todos los alineamientos son obtenidos y se ordenan por puntuación.

    b. Alineamientos quiméricos: Si el mejor alineamiento no cubre a toda la lectura, se realizan conexiones quiméricas con otras ventanas que pueden estar a grandes

distancias en la misma hebra, diferente hebra del mismo cromosoma o incluso diferentes cromosomas. Puede incluso encontrarse alineamientos quiméricos donde las lecturas apareadas pertenecen a distintas ventanas, ayudando a detectar la posición exacta de la unión de la quimera.

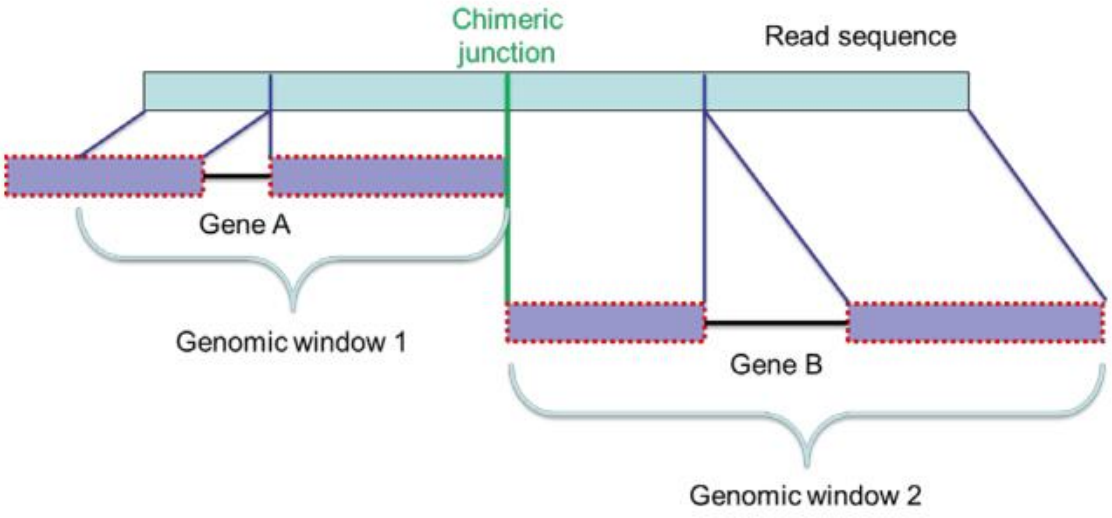


Figura 20. Ejemplo de una unión quimérica. Algunas lecturas mapean perfectamente en la primera región, otras en la segunda y algunas cruzan la unión quimérica entre los exones de los dos genes.

La unión de fragmentos se guía siguiendo un modelo de puntuación como los de los algoritmos de alineamiento local, con penalizaciones, decididas por el usuario, tanto para errores, aciertos, inserciones, deleciones y huecos de los lugares de splicing, permitiendo una cuantificación de calidades y rangos. La combinación con la mayor puntuación es la que se seleccionará.

$$S = +\sum_{match} P_m - \sum_{mismatch} P_{mm} - \sum_{inserion} P_{ins} - \sum_{deletion} P_{del} - \sum_{gap} P_{gap}$$

Ecuación 2. Cálculo de puntuación del alineamiento, que es la suma de las posiciones que casan (*matches*) menos la suma de las posiciones no emparejadas (*missmatches*), inserciones, deleciones y una penalización por la existencia de gaps.

STAR tiene validación experimental ya que sirvió para describir y encontrar nuevos sitios de splicing, así como localizaciones quiméricas en los genes (como el transcrito fusionado BCR-ABL en las células K562 de eritroleucemia). Requiere gran memoria RAM, pero es muy rápido: esto es debido a la indexación en matrices no comprimidas.

2.3.2 Salmon

Es un método muy interesante ya que es tan rápido como STAR pero los requerimientos de memoria son mucho más pequeños, al no producirse un alineamiento *per sé*. Además, permite corregir por el sesgo de GCs, mejorando en gran cantidad la precisión en la estimación de las abundancias. Utiliza un

algoritmo muy ligero que como entrada recibe un transcriptoma de referencia (que puede generarse fácilmente a partir de un genoma y el gtf) y unas secuencias de lecturas, pero no las alinea en su totalidad. De hecho, Salmon es capaz de realizar tanto el mapeo como la cuantificación, pudiéndose solo usar para cuantificación en el caso en que ya dispongamos de BAM para utilizar como input. Salmon añade alguna novedad respecto a los algoritmos ligeros contemporáneos: utiliza sesgos específicos de cada muestra, como el sesgo de GCs, sesgos que tienen en cuenta la posición para la cobertura de una región, los sesgos de secuenciación en los extremos 5' y 3' de los fragmentos, la distribución de fragmentos de distintas longitudes e incluso el uso de métodos específicos de cadena. Esto es algo muy positivo ya que la gran variabilidad entre replicados biológicos es uno de los grandes problemas del análisis de RNA-seq y que, al ignorarse, suelen producir muchos falsos positivos en estudios de expresión diferencial[4]. El algoritmo de Salmon es capaz de aprender estos sesgos específicos de muestra y utilizarlos para las estimaciones de abundancia de transcritos. Cuando se produce el alineamiento de las lecturas a una referencia que comparte bastantes subsecuencias, una única lectura tendrá muchas posiciones donde podrá alinear con la misma o casi exacta precisión, y tener todas esas posiciones en cuenta tiene grandes efectos en los análisis que se desarrollen posteriormente en el pipeline [48], y tener que contar con esta información hace que los análisis sean mucho más lentos y menos eficientes. Sin embargo, sabemos que toda esta información no es necesaria en determinadas condiciones: si quieres analizar los transcritos, con simplemente saber cuál es el transcrito y las posiciones en las que una correspondiente lectura mapea es suficiente.

Salmón, en vez de utilizar como aproximación el uso de *k-mer*, crea un índice FM, que puede utilizarse para determinar el número de veces que aparece una determinada secuencia en un texto comprimido, así como saber la posición de esta ocurrencia, lo cual sería información suficiente para lo que estamos interesados. Este índice se crea a partir de la transformada de Burrows-Wheeler. Para los autores de Salmon, la pieza clave para una cuantificación eficiente es saber las localizaciones de cada una de las lecturas: el alineamiento óptimo no es necesario mientras se disponga de la puntuación de un alineamiento concreto. Para ello, Salmon utiliza un alineamiento ligero basado en encontrar cadenas de secuencias exactas máximas (MEMs) o secuencias super exactas máximas (SMEMs). Para construir el índice, utiliza una matriz de sufijos junto al índice FM y así encontrar subsecuencias de cualquier longitud entre la lectura y el transcriptoma. Si la lectura mapea perfectamente en toda su longitud, utilizando los índices podremos rápidamente obtener su localización. Si hay errores o variaciones entre la referencia y la muestra, no podemos esperar que todas las lecturas mapeen en toda su longitud sin errores. Para solucionar esto, se hacen uso de las cadenas de MEMs: se obtiene el set de coincidencias exactas de mayor longitud entre la lectura y el set de transcritos. Cada MEM es una secuencia que presenta su posición respecto a la lectura y su posición respecto al transcriptoma. Posteriormente, estos MEM pueden encadenarse para formar apareamientos aproximados de mayor longitud entre la lectura y los transcritos: son cadenas de secuencias que alinean perfectamente interrumpidas por errores aislados o pequeños indels.

Su principal ventaja es que mantiene la rapidez de otros alineadores como STAR pero además no es tan extensivo computacionalmente, al no almacenar realmente el alineamiento. Por ese motivo, sus requerimientos de memoria RAM son mucho más accesibles para poder utilizarse en cualquier ordenador personal.

2.4 Reconstrucción de isoformas y cuantificación

A este complejidad inicial del alineamiento de las lecturas del ARN-seq, debemos sumarle la posibilidad de que por el splicing alternativo se pueden producir distintas isoformas, muchas de las cuales comparten sus exones, dificultando que, incluso una vez realizado el alineamiento de la lectura en su correspondiente exón/región del ARNm maduro, no sepamos a cuál de todas las isoformas en las que esa región está presente pertenece esa lectura en concreto. Para procesar toda esta información se utilizan complejos modelos matemáticos y estadísticos en lo que se conoce como reconstrucción de isoformas. Es un paso fundamental y limitante en los métodos de estudio del análisis de splicing diferencial, debido a que en la actualidad la longitud de las lecturas utilizadas en la secuenciación sigue siendo inferior a la longitud total media de los transcritos, por lo que muchas de las pequeñas lecturas serán compartidas por varias isoformas.

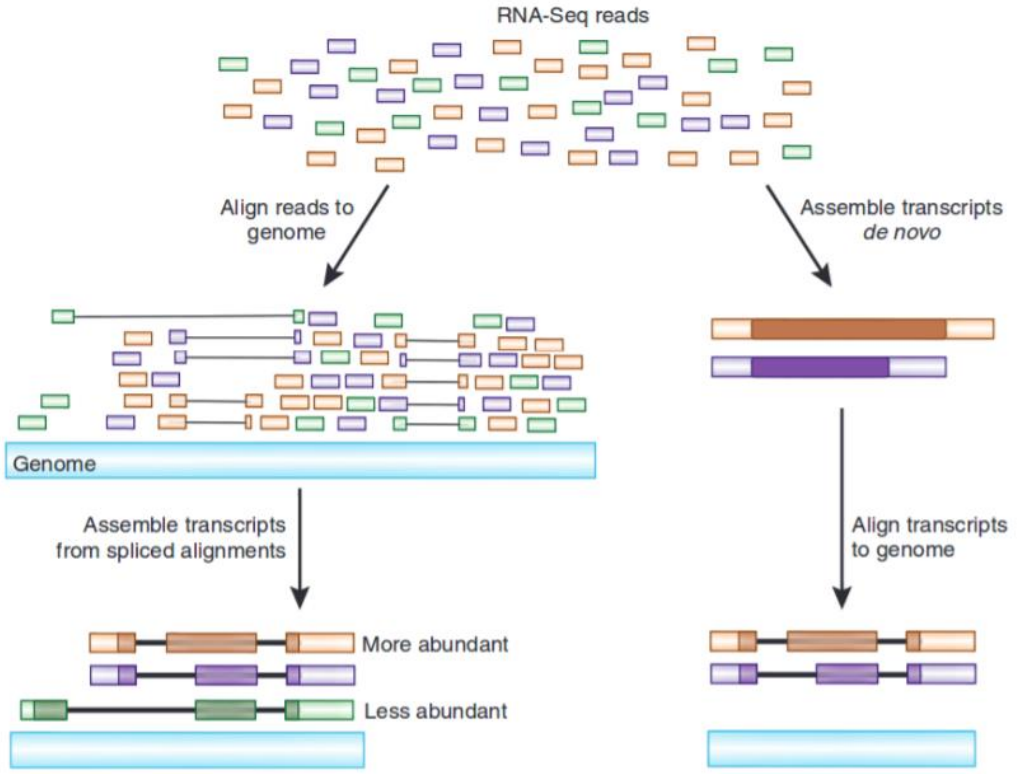


Figura 21. Procesamiento de las lecturas para obtener las diferentes isoformas de los ARNm maduros. Posterior al alineamiento de las lecturas, es necesario ensamblar los transcritos de los que provienen, ya que muchas de ellas son idénticas, pero pertenecen a distintos transcritos que comparten algún exón. Haas, B. J. & Zody, M. C. Advancing RNA-Seq analysis. *Nat. Biotechnol.* 28, 421–423 (2010).

Existen varios métodos para la reconstrucción del transcriptoma: guiados por genoma o independientes de genoma (Figura 22) [46]. Los dependientes de genoma primero mapean todas las lecturas a la referencia y después ensamblan las lecturas que solapan en transcritos, mientras que los independientes de genoma ensamblas las lecturas directamente en transcritos sin necesidad de ninguna referencia.

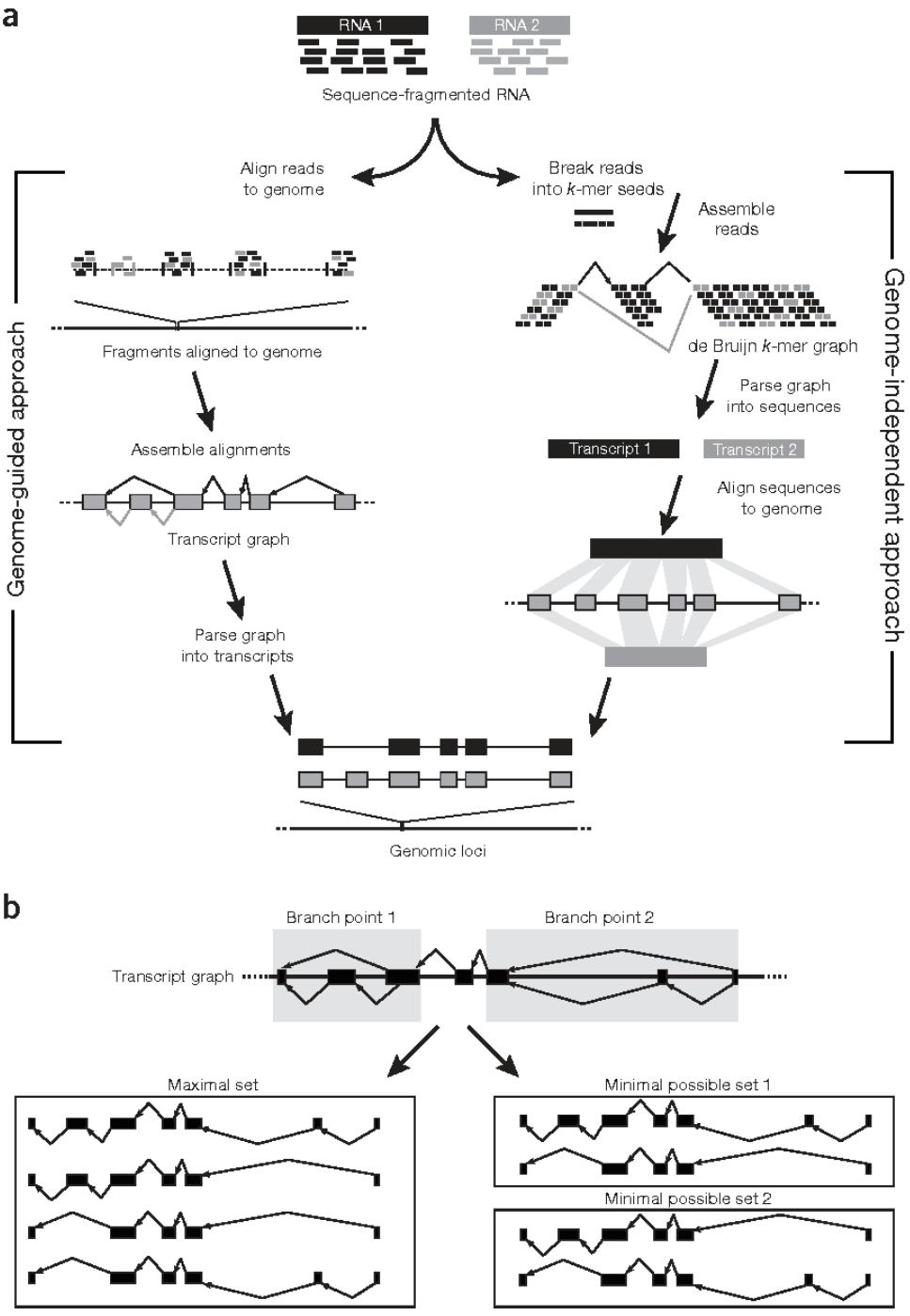


Figura 22. Métodos de reconstrucción de isoformas. (**a**) Las lecturas que proceden de dos isoformas diferentes están en distinto color (negro y gris). En el ensamblaje guiado por genoma, las lecturas son inicialmente mapeadas al genoma, y las lecturas que caen en regiones de splicing utilizadas para construir un gráfico de transcritos, que posteriormente es transformado en anotaciones de genes. En los métodos independientes de genoma, las lecturas se rompen en semillas de k-mer y se ponen en un gráfico de Bruijin. El gráfico se recorre para intentar identificar las secuencias del transcrito. (**b**) Las lecturas de splicing permiten cuatro posibles isoformas, pero solo dos de ellas son necesarias para explicar todas las lecturas existentes.

2.4.1 RSEM

RSEM (ARN-seq por esperanza-maximización)[5] consiste en varias iteraciones del algoritmo EM (Esperanza-Maximización) para asignar cada lectura a la isoforma de la que provienen. Tiene la opción de utilizar como alineador interno bowtie2 o STAR, o directamente pasarle unos archivos BAM/SAM que cumplan las condiciones especificadas por los creadores; entre estas destaca el hecho de que el alineador debe reportar todos los sitios válidos de mapeo para cada lectura, para que sea RSEM y sus algoritmos los que determinen cuál es la posición más adecuada y asegurar una mayor precisión. Este programa devuelve estimaciones a nivel de genes y a nivel de isoformas mediante la abundancia por máxima verosimilitud basado en EM tanto como número de fragmentos y fracción de los transcritos dada una isoforma o gen. También presenta una herramienta que permite ver la visualización del alineamiento y la cantidad de lecturas que caen en cada región utilizando el buscador genómico UCSC. Permite especificar el origen de las cadenas según el protocolo seguido y permite indicar si por la tecnología utilizada las lecturas presentarán una mayor distribución de sesgo en los extremos 5' o 3'. Es uno de los métodos que es independiente de genoma de referencia, sino un transcriptoma. Consiste en dos pasos:

1. Generación y preprocesado de los transcritos de referencia (*rsem-prepare-reference*)
2. Alineamiento de las lecturas a los transcritos de referencia siguiendo la estimación de las abundancias y sus intervalos de confianza utilizando el algoritmo EM.
   El modelo estadístico de RSEM (Figura 23) permite modelar tanto las lecturas apareadas (R^1 y R^2) como single-end, ya que R^2 actúa como una variable latente en ese caso. Además, para modelar la longitud de los fragmentos de los que deriva una lectura se utiliza la variable F. A su vez, la longitud de las lecturas se representa por la variable L. La Q representaría las puntuaciones de esas lecturas en el caso en el que se presentasen en formato Fastq. El error se mide mediante una función empírica: $P(r_i|q_i, c)$: probabilidad condicional de la lectura r en la posición i determinado por la calidad en esa posición (q_i) y deriva del nucleótido de referencia c. Si no se presentan calidades, se utiliza un modelo generativo de error dependiente de posición y referencia[49].

Para alimentar el modelo gráfico, necesitamos introducirle los parámetros en forma del vector θ, que son las probabilidades, para cada fragmento, de que deriven de un determinado transcrito i. Para ello, RSEM utiliza el algoritmo de EM. Como puede verse en la Figura 24 [50], el uso del algoritmo de EM para la cuantificación de transcritos consiste en varios ciclos consecutivos donde se va a ajustando la proporción de cada isoforma. Se sabe que este proceso siempre termina en convergencia: parará cuando las probabilidades de que todos los fragmentos deriven de ese transcrito en concreto tengan un cambio relativo menor que 10^{-3}. En el paso E (Esperanza) se estima la distribución de las variables ocultas de los datos (es decir, saber de qué clase, transcrito en este caso, provienen) en función de los valores actuales de los parámetros, y asignamos cada punto (lectura) a su clúster más cercano (isoforma). En el paso M (maximización) se vuelve a calcular la proporción de cada una de las isoformas y se maximiza la unión entre la distribución de los datos y la clase a la que pertenecen. De nuevo se repite el paso E con esas nuevas probabilidades actualizadas, hasta que converge el resultado.

Además del estimador de máxima verosimilitud, RSEM permite el calculo de la versión Bayesiana del modelo, obteniendo el 95% de los intervalos de confianza y la probabilidad posterior (PMEs, *posterior mean estimate*) utilizando Gibbs-sampling.

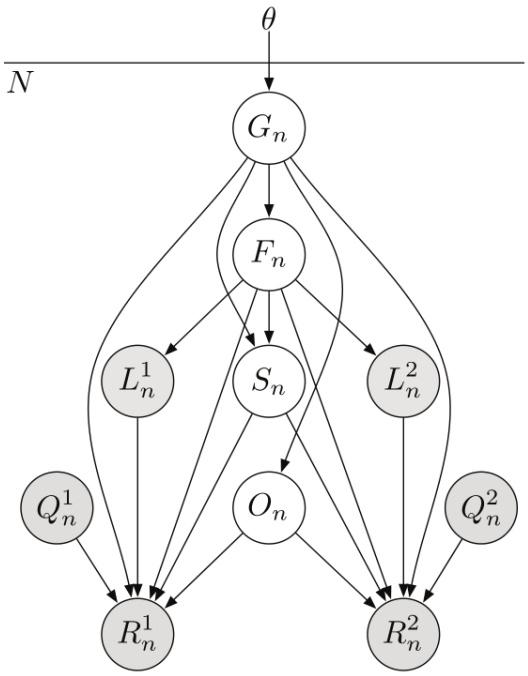


Figura 23. Modelo gráfico dirigido de RSEM. N es el número de variables aleatorias, que representan el número de lecturas secuenciadas. Para el fragmento n, la longitud, posición de inicio, orientación y transcrito del que provienen se representa como Fn, Sn, On y Gn, respectivamente. Los parámetros que utiliza el modelo se dan en el vector θ, que son las probabilidades a priori de que cada fragmento derive de cada uno de los transcritos.

2.4.2 Salmon

Como hemos descrito anteriormente, Salmon es capaz de estimar el porcentaje de cada una de las isoformas tanto a partir de alineamientos tradicionales como de *quasi-alineamientos*. Tanto si partes de un BAM como de las lecturas, al final podrás obtener de Salmon una matriz que presenta la proporción de cada una de las isoformas.

Para esto, Salmón utiliza aproximaciones “online” y una “offline”. En la fase online permite estimar los niveles de expresión iniciales, los parámetros auxiliares, los modelos de sesgos y asociar cada fragmento con su isoforma, mientras que en la offline se refinan estas estimaciones iniciales.

- Fase online: Utiliza, para la inferencia, una variante del modelo de inferencia bayesiana estocástico colapsado variacional [51], que utiliza la Asignación Latente de Dirichlet (ALD) adaptado a datos de grandes dimensiones.
- Fase offline: Aplica el algoritmo de EM estándar o el algoritmo de EM bayesiano sobre una pequeña representación de los datos hasta alcanzar convergencia. Así obtiene las probabilidades posteriores generadas por Bootstrap o Gibbs sampling.

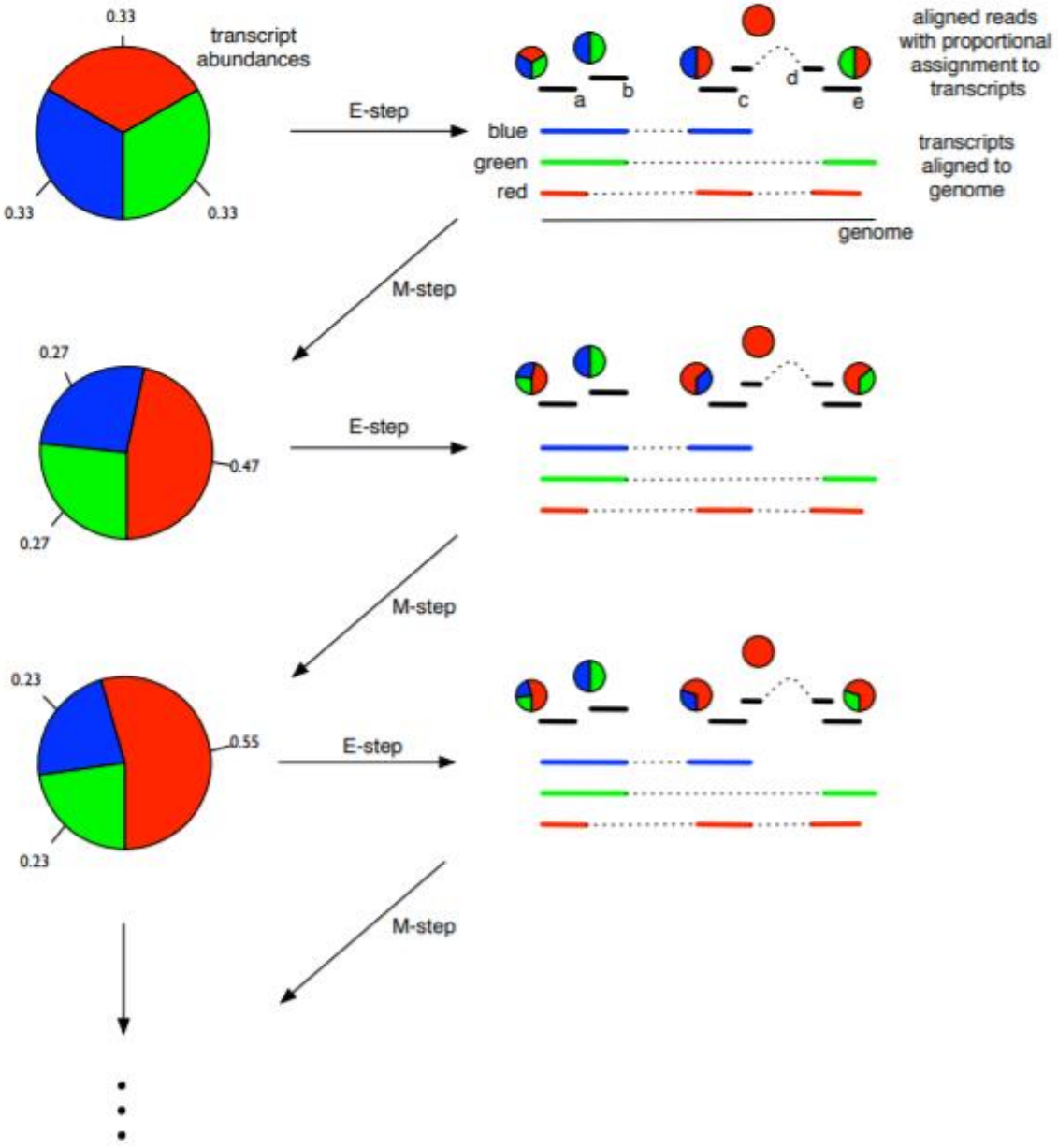


Figura 24. Algoritmo de EM en RNA-seq. Un gen presenta tres isoformas (azul, rojo y verde) en este caso, todas de la misma longitud (ya que la probabilidad de pertenecer a una u otra dependerá de esta variable). Inicialmente a cada isoforma se le asigna la misma abundancia. Las lecturas que disponemos (a, b, c, d, e) presenta una que mapea en todas las isoformas, una que solo mapea en la roja y las otras tres para todas las combinaciones de dos pares de las isoformas. Durante el paso de Esperanza (E), las lecturas son asignadas a un transcrito en función de las abundancias de las isoformas. Durante el pazo de maximización (M), las abundancias se recalculan en función de las lecturas asignadas.

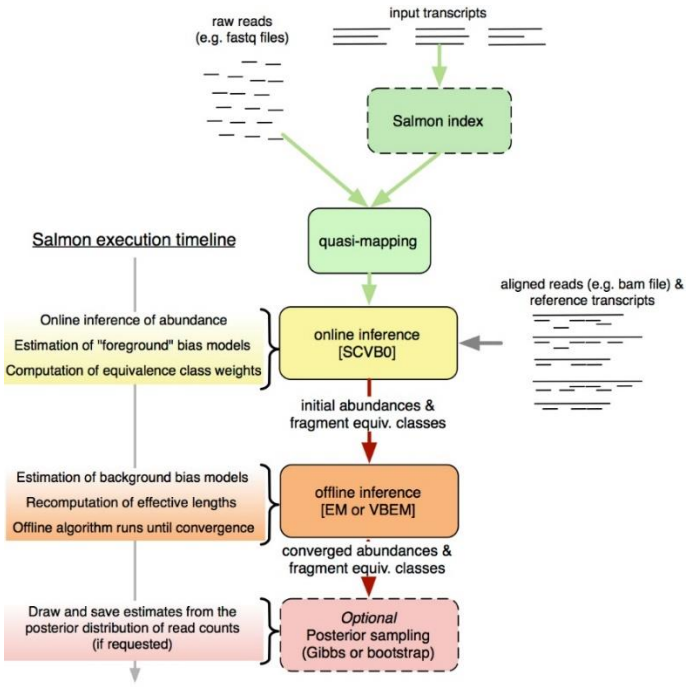


Figura 25. Resumen del método de Salmon. Puede aceptar tanto lecturas (flechas verdes) como alineamientos (flecha gris). Al procesar estos alineamientos (tanto el quasi-mapping como el BAM, Salmon ejecuta un algoritmo de inferencia online que asegura que la abundancia de los transcritos estará disponible para estimar los pesos de cada una de las clases. Después de la aportación a los estimados de abundancia y los modelos de sesgos, a cada fragmento se le asigna a su clase equivalente (o se crea si no existe)

2.5 Expresión diferencial: Análisis de splicing alternativo

Una vez hemos obtenido la cuantificación a la que pertenece cada una de las isoformas, queremos observar si alguno de los cambios en la expresión entre condiciones implica un uso diferencial de transcrito, es decir, detectar si ha habido *switching:* queremos detectar cuando ha habido un intercambio de expresión entre distintas isoformas, sugiriendo que en la segunda condición se ha producido un procesado diferencial respecto a la primera.

2.5.1 ASapp

Es un método para realizar análisis de splicing alternativo entre dos estados biológicos. Es un programa que cuenta con interfaz gráfica (Figura 27) para que los usuarios puedan introducir el input y obtener resultados. Utiliza información de Biomart y APPRIS para anotar. Para evaluar la variación en la proporción de transcritos en función de la condición a nivel de gen, utiliza como métrica la distancia de Jensen-Shannon[52], [53], que cuantifica el nivel entrópico del gen por condición. Este valor será cercano a cero en el caso en el que no haya uso diferencial de transcrito, mientras que se acercará a uno cuando exista una mayor diferencia entre condiciones. Luego utiliza un modelo de clasificación de gaussianas (GMM) sobre la distancia de Jensen-Shannon y determina la probabilidad de cada elemento de pertenecer al grupo, obteniendo así una probabilidad de splicing/no splicing para ese gen. La aplicación permite al usuario fijar dicha probabilidad. Una vez seleccionados los genes que sufren splicing alternativo, los transcritos más correlacionados con cada condición son seleccionados.

Utiliza el porcentaje de isoforma (en TPMs) respecto al gen del que provienen para cada muestra y una matriz de diseño para conocer las condiciones diferenciales de cada una de ellas. Puede utilizar la base de datos APPRIS para obtener más información acerca de las distintas isoformas afectadas y el tipo de splicing que se ha producido en cada situación, permitiendo obtener una gran cantidad de información funcional. Además, permite aplicar un filtro, si se disponen de los datos de expresión diferencial de genes, por aquellos genes que han mantenido constante su expresión, pero sí que han presentado un notable cambio en la isoforma, lo que puede ser muy relevante.

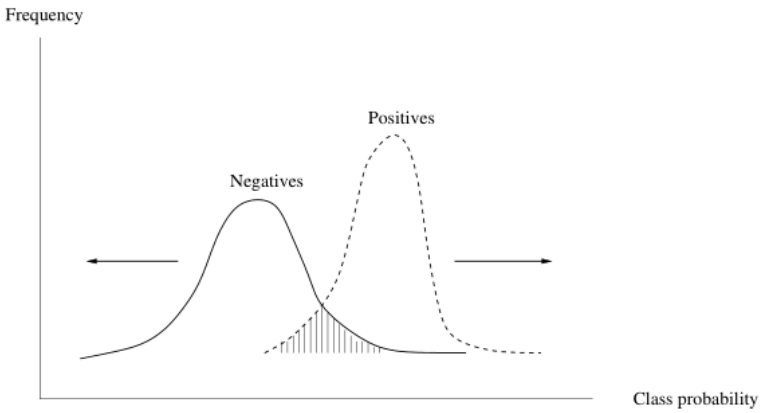
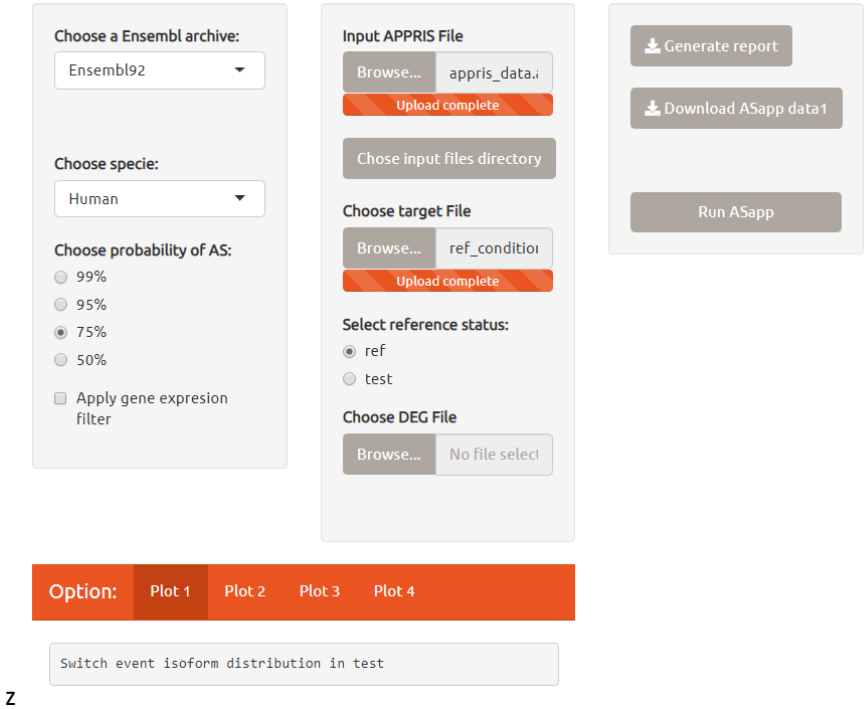


Figura 26. Modelo de mezcla de Gaussianas. Se representa la distribución de datos como una mezcla de dos gaussianas y se busca ver a que punto pertenece cada una de ellas. En las regiones que son totalmente de un tipo o de otro no hay conflicto, pero en las regiones donde las curvas solapan, es difícil establecer a que grupo pertenece cada una de las mismas.



z

Figura 27. Interfaz gráfica de ASApp

2.5.2 DRIMSeq

DRIMSeq es un soporte estadístico que utiliza la distribución multinomial de Dirichlet para encontrar cambios en el uso de isoformas entre condiciones. Es un modelo que de manera natural tiene en cuenta la expresión diferencial de genes sin perder información de la abundancia génica. Teniendo en cuenta que los genes pueden presentar diferentes isoformas, la expresión génica puede describirse como una expresión multivariada de transcritos. Por ese motivo, DRIMSeq utiliza la distribución Dirichlet, que es una versión multivariada de la distribución beta (distribución continua con valores entre 0 y 1 y que presenta dos parámetros, α y β, que definirán la forma de la función de densidad de la probabilidad). Como ya hemos visto, la distribución de Poisson es una distribución univariada incapaz de captar la varianza de los datos de expresión génica de ARN-seq ya que la media es igual a la varianza y una extensión de esta es la binomial negativa, que se utiliza para casos de pocas muestras (y, por ende, gran varianza) y que se utiliza en gran cantidad de aplicaciones genómicas, como en *Polyester*. De la misma manera, cuando se busca modelar datos multivariados, la aplicación básica es la distribución multinomial, que modela proporciones entre varias características. Para permitir una mayor variabilidad, se suele utilizar la distribución multinomial de Dirichlet, permitiendo ser más robusto con menos replicados.

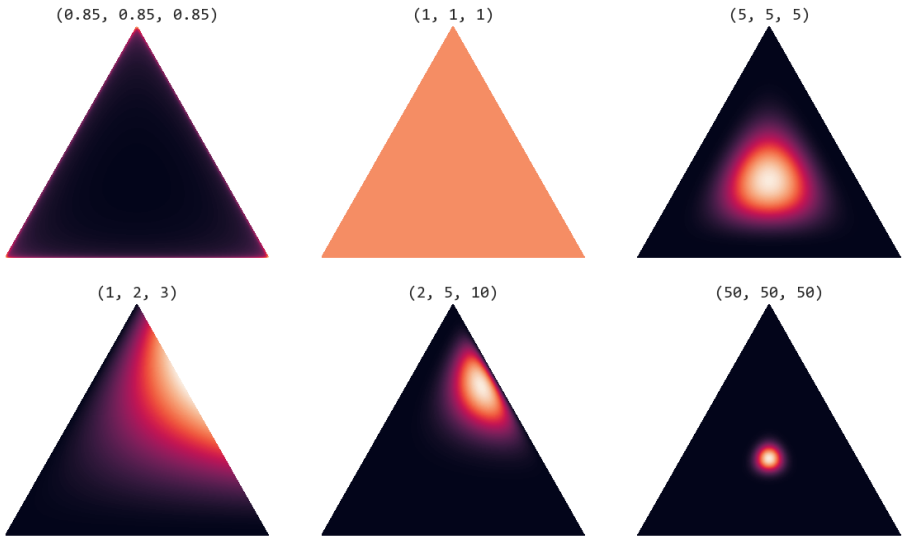


Figura 28. Representación de la distribución de Dirichlet con 3 muestras para distintos valores de α. Se ve que alfa controla la fuerza: (1,1,1) es una distribución uniforme. (50,50,50) es una distribución muy estrecha pero centrada, etc. Cuando alfa es menor que uno para todas las condiciones, se obtienen picos en las esquinas.

2.5.3 SUPPA2

Es un método muy completo para el estudio del splicing alternativo. Como comentábamos en la introducción, existen muchos abordajes para intentar comprender y estudiar el mecanismo de splicing alternativo, a nivel de uso diferencial de transcrito, uso diferencial de exones e incluso a niveles de eventos de splicing que se han producido, y SUPPA2 permite tratarlos a todos.

Por este motivo, para la cuantificación de transcritos calcula los valores de inclusión (PSI) de los distintos eventos de splicing alternativos para cada muestra. SUPPA2 considera la existencia de dos distribuciones: una donde la variación en los valores de inclusión entre replicados biológicos y otra entre condiciones, permitiendo simular bien la variabilidad biológica y la variabilidad entre condiciones. Para clasificar las muestras, utiliza dos modelos basados en densidad: DBSCAN y OPTICS, siendo el último una optimización del primero, que genera mejores resultados, pero requiere más tiempo. Este tipo de modelos no necesitan la especificación del número de clúster. Ambos métodos utilizan los vectores de valores medios de PSI (niveles de inclusión) por evento y requieren que se indique el número mínimo de eventos por clúster. Para calcular este valor PSI, dadas las abundancias de todas las isoformas de los transcritos en TPM. Siendo F1 y F2 los casos opuestos para un evento concreto (por ejemplo, siendo el evento exclusión de exón, el F1 sería los que lo incluyen y el F2 los que lo saltan), el valor de psi es la ratio entre los TPMs de las isoformas que lo incluyen entre los TPMs de las isoformas que lo excluyen.

$$\Psi = \sum_{k \in F_1} \mathrm{TPM}_k / \sum_{j \in F_1 \cup F_2} \mathrm{TPM}_j$$

3 Sistema y diseño

3.1 Diseño

Para realizar acabo todo el proceso de *benchmarking*, es necesario establecer unas condiciones iniciales para su estudio, delimitar unos pipelines concretos y seleccionar el proceso a seguir. Es uno de los pasos más importantes ya que nos permitirá establecer unos valores óptimos para realizar una estandarización del estudio de este proceso.

3.1.1 Selección de las variables

En la generación de lecturas, era necesario escoger las distintas condiciones a estudiar.

- **Elección de la longitud de las lecturas:** El objetivo es evaluar, para diferente longitud de transcritos, la eficiencia y capacidad de cada uno de los pipelines. Los primeros análisis que supondrán el desarrollo de este trabajo se quieren realizar en condiciones no limitantes, viendo que en el caso de RNA-seq es suficiente con utilizar unas lecturas de más de 50 pares de bases es suficiente, mientras que para splicing alternativo puede ser recomendable que esta longitud sea algo mayor[54]. Debido a que el objetivo es diseñar un pipeline que se pueda seguir de forma rutinaria en los laboratorios para incrementar los estudios de splicing alternativo, el objetivo es obtener las mejores condiciones eficiencia/precio posibles[55]. Se sabe que a partir de los 60 pb no hay mucha diferencia en la eficiencia, por lo que se utilizarán longitudes de 60, 75 y 100 pb para evaluar la capacidad de cada uno de los métodos, partiendo de la condición menos limitante de longitud de 100 pb.

	Precio por canal	
Tamaño de las lecturas	Single End	Paired end
25 pb	950 $	1275 $
50 pb	1100 $	1650 $
75 pb	1250 $	2025 $
100 pb	1400 $	2400 $

Tabla 1. Precio de las lecturas en función de la longitud de las mismas y el método escogido.

- **Elección de cobertura:** Durante muchos años se ha estudiado el efecto del incremento de la profundidad de lectura en la detección de genes diferencialmente expresados como puede verse en la Figura 29**a/b**. Se observa que, a partir de los 10 millones de lecturas, la equivalencia entre el incremento de precio y el incremento de mejora no será gran cosa, ya que la función tiende a ser constante a partir de ese punto [56], [57]. En el caso concreto de splicing, se ha visto que es recomendable que esta profundidad sea un poco mayor, ya que permite identificar mejor los sitios de unión de splicing, estandarizando una cantidad aproximada de entre 50-100 millones de lecturas (Figura 29**b**)[58].

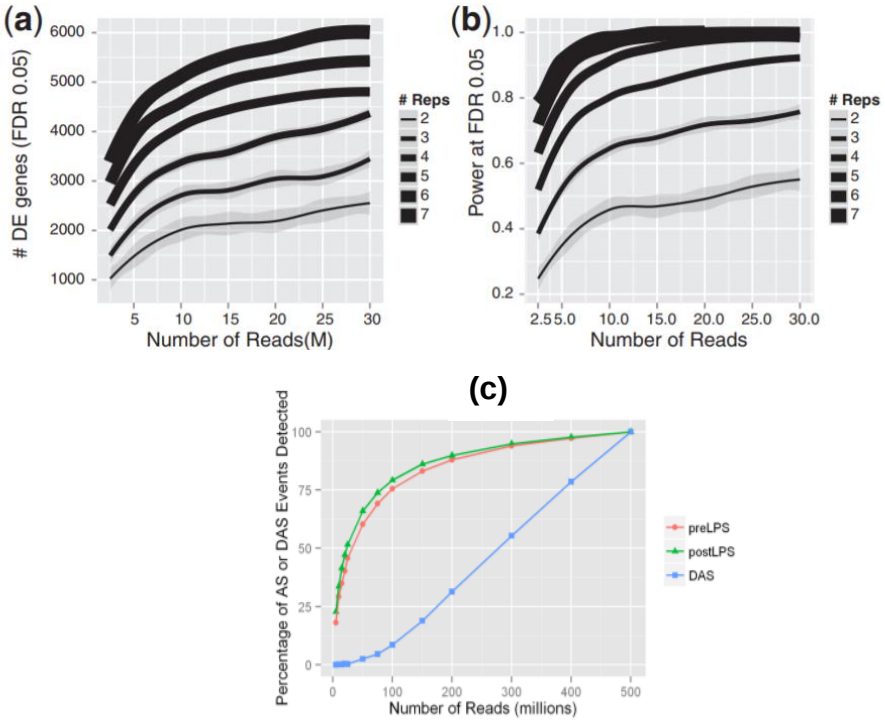


Figura 29. Variación de (**a**) la sensibilidad (genes diferencialmente expresados) y el poder de detección de genes diferencialmente expresados (**b**) entre distinto número de replicados y distinta profundidad de lectura. (**c**) Variación en el porcentaje de eventos de splicing detectados en función del número de lecturas de la secuenciación.

Para el desarrollo de nuestras lecturas sintéticas vamos a realizar una estimación de la profundidad de estas a partir de una proporción de cambio en la cobertura, *fold coverage,* que representa el número de veces que aparece cada base en las lecturas, y que, por ende, para cada transcrito, será dependiente de la longitud de este. Para ello, calcularemos el valor que necesitamos en función de cada una de las condiciones elegidas[59].

$$Fold\ coverage = \frac{N * L}{G}$$

Ecuación 2. Ecuación para el cálculo del fold coverage necesario para cada longitud de lectura (L) y tamaño de la referencia (G) según el número de lecturas finales que queramos (N)

Eligiendo una profundidad adecuada para splicing de aproximadamente 60 millones de lecturas, y fijando la longitud de las lecturas en 100 pb como condición no limitante:

$$Fold\ coverage = \frac{6 \cdot 10^7 * \ 100}{3{,}23639784 \cdot 10^7} \approx \ 18X$$

Seleccionamos 20X como profundidad no limitante para testar nuestras diferentes lecturas. Para el cálculo de este valor, hemos necesitado obtener el tamaño del transcriptoma que hemos generado, y de esa versión concreta. En vez de utilizar valores aproximados, se desarrolló un sencillo script en Python que, introduciendo el transcriptoma en formato fasta, nos permite calcular tanto la longitud de este como el número de transcritos que presenta (Figura 30).

```
def count_tx_len(fasta):
    transcriptome_tx = 0
    transcriptome_len = 0
    with open(fasta) as all_transcripts:
        for line in all_transcripts:
            if line[0]=='>':
                transcriptome_tx +=1
            else:
                transcriptome_len+=len(line.strip())
        return transcriptome_tx, transcriptome_len
```

Figura 30. Función que permite el cálculo del número de lecturas del transcriptoma de referencia, y de su tamaño total.

- **Número de réplicas biológicas**: Es un factor muy importante ya que, como hemos visto en la Figura 29**a** y **b**, supone una mejora sustancial en el poder de detección, pero que, a su vez, supone un incrementa muy grande del precio, ya que deberíamos multiplicar los precios previamente descritos por el número de replicados que deseamos introducir. Observando esta segunda gráfica, podemos ver que a partir de 6-8 replicados la gráfica comienza a tornarse como una constante, reduciéndose la capacidad de mejora, aunque es sustancialmente superior a la que conlleva el incremento de la profundidad [60]. Este motivo nos permite establecer como condición inicial el número de seis réplicas para nuestra simulación.

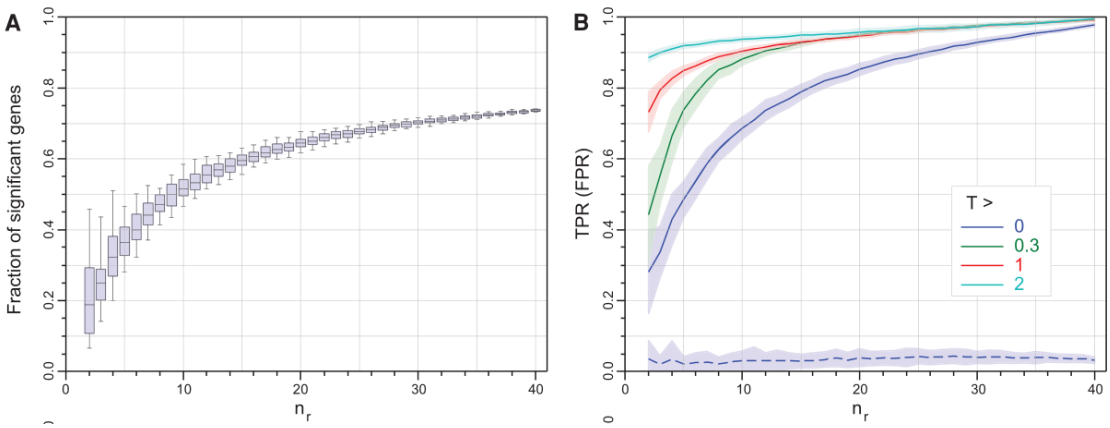


Figura 31. Porcentaje de genes significativos detectados en función del número de replicas, n_r.

- Desarrollo de la matriz de porcentaje de cambio (*fold change*): Como hemos explicado, *Polyester* permite indicar, para cada gen, la proporción de cambio que presenta entre una condición y la siguiente. Con el objetivo de permitir más estudios y analizar el funcionamiento de cada uno de los pipelines elegidos en función de cómo sea la diferencia entre las dos condiciones (capacidad de detectar pequeñas diferencias, problemas de falsos positivos, falsos negativos…) realizamos la generación de una matriz de *fold change* progresiva. Tras mezclar la lista de transcritos de referencia, para evitar que todas las isoformas del mismo gen se encuentren en la misma región de la matriz, la dividimos en distintos fragmentos: El 70% de ella no mostrará ninguna diferencia entre condiciones. Esta cantidad fue decidida para simular lo máximo posible los datos biológicos, ya que los programas tienen en cuenta que la mayoría de las secuencias no sufren este cambio por lo que no tenerlo en cuenta sería sesgarlo. Este dato se obtuvo a partir de las referencias[61].
- Versión del genoma: A la hora de anotar los transcritos que utilizaremos, de la versión del genoma GRCh38.p12, se utilizó la última versión disponible de gencodev30, ya que se incrementaba el número de transcritos anotados para analizar, siendo especialmente importante la incorporación de nuevas isoformas.
- El abordaje inicial implica el uso de lecturas *single end*.

3.1.2 Selección de los programas

A la hora de seleccionar los programas a estudiar se tuvieron en cuenta tres aspectos relevantes: fiabilidad del método a utilizar, velocidad del pipeline en su conjunto y requerimientos del sistema.

Por este motivo, como alineadores se seleccionaron como alineadores STAR y Salmon, ya que son dos de los métodos más punteros. El primero cuenta con un novedoso método de indexado que le permite tener gran velocidad, pero para su ejecución fue necesario enviarlo a un clúster para que tuviese suficiente RAM. Por otro lado, Salmon no genera realmente el alineamiento, pero aporta la información necesaria para poder cuantificar los transcritos, esto le concibe mucha velocidad y se encuentra entre los métodos preferidos por su facilidad de uso, pudiéndose correr en un ordenador de manera eficiente y rápida.

La cuantificación después del alineamiento por parte de STAR viene de la mano de RSEM, uno de los algoritmos que mejor es capaz de estimar y reconstruir las isoformas. Por su parte, Salmon también se

encarga de la cuantificación, ya que utiliza sus propios datos de *quasi-alineamiento* para ello, completando el pipeline con el análisis diferencial. Este análisis diferencial lo realizaran tres programas: DRIMSeq, con su uso de la distribución de Dirichlet, es uno de los métodos más novedosos, y por ese motivo poco testado, que permite realizar un análisis funcional del splicing alternativo de manera eficiente a partir de la línea de comandos, permitiendo incluso obtener gráficas directamente en su implementación. El uso de ASapp permite un análisis más sencillo por medio de interfaz gráfica, filtrándote directamente los resultados que son más importantes en función del nivel de astringencia proporcionado por el usuario, generando todas las gráficas al momento. Por último, se añade SUPPA al estudio debido a su gran uso en el estudio de eventos de splicing, para testar su nueva adaptación incorporada para la detección de DTUs.

4 Proceso, experimentos y resultados

4.1 Generación de las lecturas

Para la generación de las lecturas, elaboramos un script que permita ir generando las lecturas en distintas condiciones en función de las variables que deseamos ir modificando para estudiar distintas condiciones: la longitud de las lecturas, la cobertura, si las lecturas estarán emparejadas y el número de replicados.

```
karu@Karutsuki:~/share$ Rscript simulate_experiment_arguments_bash.R --help
Usage: simulate_experiment_arguments_bash.R [options]


Options:
        -l READS_LENGHT, --reads_lenght=READS_LENGHT
                Length of the reads

        -c COVERAGE, --coverage=COVERAGE
                Coverage

        -p PAIRED, --paired=PAIRED
                True if paired, false if single-end

        -f FILE, --file=FILE
                full path to the RNA transcripts fasta file

        -r REPLICATES, --replicates=REPLICATES
                Number of replicates you want to optain. Will be created one by one

        -a ALL, --all=ALL
                If true, in each round makes both, singled and paired ended

        -h, --help
                Show this help message and exit
```

Figura 32. Visión de la interfaz del script que permite simular lecturas en función de las condiciones deseadas.

Este código llama de manera interna a *Polyester* y nos permitirá generar las lecturas con unas condiciones fijadas. Con el parámetro de cobertura, *coverage*, que presenta la profundidad de la

lectura en forma de fold coverage, para calcular la cantidad de lecturas en función de la longitud del ARN de referencia (rna_fasta) y la longitud de lectura escogida, que se pasará por la línea de comandos.

```
#OBTAIN THE IDENTIFIERS FROM THE TRANSCRIPTS NAMES

names_tx <- names(rna_fasta)
chunks_names <- strsplit(names(rna_fasta), ' ')
simple_names_tx <- lapply(chunks_names, function(l) l[[1]])

## READS PER TRANSCRIPT

length_tx <- matrix(round(width(rna_fasta)), ncol = 1)
rownames(length_tx) <- simple_names_tx

readspertx <- matrix(round(args$coverage * width(rna_fasta) / args$len), ncol = 1)
rownames(readspertx) <- simple_names_tx
```

Por último, otro de los parámetros fijos es la matriz de cambio o *fold change*. Para ello, presentamos cinco condiciones: i) Que no haya expresión diferencial, y por tanto ambas condiciones tendrán el valor 1 de fold change; ii) que la segunda condición (test) esté infraexpresada respecto a la condición de referencia (*second_chunk*), iii) que la segunda condición esté un poco sobreexpresada, con un fold change entre 1 y 2, IV y V) Que esta sobreexpresión sea más exagerada, con unos valores entre 2 a 8.

```
## CREATE THE FOLD_CHANGE MATRIX. WE HAVE 5 CONDITIONS.

# first_chunk <- seq(1,1.1, length.out = 100)
second_chunk <- seq(0,1, length.out=200)
second_chunk <- second_chunk[!(second_chunk %in% 0)] ##We exclude 0 count reads

third_chunk <- seq(1,2, length.out =200)
fourth_chunk <- seq(2,4, length.out = 400)
fifth_chunk <- seq(4,8, length.out = 800)


fold_change_1 = matrix( rep(1,2*length(rna_fasta)), nrow=length(rna_fasta))
rownames(fold_change_1) <- simple_names_tx
```

Finalmente, realizamos una divisón de la matriz de fold_change de tal manera que la condición de no cambio (1 en ambas condiciones) represente un 70% (7 de 10), y el resto de condiciones descritas anteriormente, un 15% para cada una (1.5 de 10). Para automatizar este proceso, es necesario que el tamaño de las divisiones sea función de la cantidad total de transcritos de la muestra a analizar (filas de la matriz), que se representa con la variable length_tx.

```
size_divisions = round(dim(length_tx)[1]/10)
div_1 =  fold_change_1[1:(size_divisions*7),]
div_01 =  fold_change_1[(size_divisions*7+1):round(size_divisions*8.5),]
div_12 =  fold_change_1[round((size_divisions)*8.5+1):round((size_divisions*9)),]
div_24 =  fold_change_1[round((size_divisions*9)+1):round((size_divisions*9.5)),]
div_48 =  fold_change_1[round((size_divisions*9.5)+1):nrow(fold_change_1),]
```

Para asegurar la aleatoriedad del proceso de selección de fold_change, para cada uno de estos tramos se tomará al azar un número perteneciente a una secuencia entre los valores indicados.

```
div_01[,2] <- apply(div_01[,2, drop=FALSE], c(1,2),
                    function(fold, chunk = second_chunk) fold=round(sample(chunk, 1),4))
div_12[,2] <- apply(div_12[,2, drop=FALSE], c(1,2),
                    function(fold, chunk = third_chunk) fold=round(sample(chunk, 1),4))

div_24[,2] <- apply(div_24[,2, drop=FALSE], c(1,2),
                    function(fold, chunk = fourth_chunk) fold=round(sample(chunk, 1),4))

div_48[,2] <- apply(div_48[,2, drop=FALSE], c(1,2),
                    function(fold, chunk = fifth_chunk) fold=round(sample(chunk, 1),4))

fold_change_mx <- rbind(div_1, div_01, div_12, div_24, div_48)
```

Por último, además de los parámetros que pueden ser determinados por el usuario, existen unos parámetros prefijados que se producirán en todas las simulaciones y replicados. Para la distribución, hemos elegido la opción empírica. Este parámetro permite determinar de que distribución se toma la longitud de los fragmentos, estando en este caso determinada por una distribución estimada de los datos de Geuvadis. Respecto al modelo de error, hemos seleccionado de nuevo uno basado en los sesgos empíricos de Illumina, en concreto *illumina5,* que tiene en cuenta que el centro del transcrito es más probable que sea secuenciado.

Para agilizar el proceso, se seleccionó únicamente el cromosoma 1, ya que es uno de los cromosomas de mayor tamaño y nos permitirá testar la calidad de cada método sin necesidad de extenderse en tiempo de computación y memoria. Para ello, se obtuvo un archivo .gtf filtrado, únicamente con los transcritos del cromosoma 1, y se obtuvo un archivo .fasta con todos los transcritos, para facilitar el trabajo de RSEM.

4.2 Alineamiento

Se desarrollaron scripts que permitían correr, de forma automatizada para todas las réplicas y futuros experimentos, ambos alineadores.

- STAR: Debido a su gran requerimiento de memoria, se adaptó un pipeline de trabajo al clúster de computación, permitiendo su palarelización en distintos núcleos. Cabe destacar la importancia de la utilización de la --quantMode TranscriptomeSAM para que devuelva una salida que sea utilizable como entrada para RSEM, que requiere que se muestren todas las opciones de alineamiento posibles para seleccionar internamente cuál es la mejor posibilidad.
  Tiempo de ejecución/muestra: 5 min
  El porcentaje de lecturas no mapeadas por muestra es cercano al 0% en todas ellas.
- Salmon:
  Es importante la selección de un tamaño de k-mer adecuado para la correcta formación del índice de *quasi-mapeo*. Se ha demostrado que para lecturas de 75pb o más, un k de 31 es lo más adecuado, siendo correspondiente disminuirlo para lecturas más cortas[4]. El protocolo de generación de lecturas de *Polyester* es *unstranded*, es decir, no sabemos de dónde proviene la lectura que obtenemos. Por ello, le indicamos estas condiciones a salmón mediante el uso del tag -l IU (inward unstranded). Tras la construcción del índice, que solo será necesario una vez para todas nuestras lecturas generadas, y debido a la rapidez y pocos requerimientos computacionales de este método, se desarrolló un script que permite detectar todas las lecturas que se encuentran en un determinado directorio y correr secuencialmente el programa con ellas, generando los outputs deseados con los mismos nombres en la carpeta en la que se desee.

Se aprovecha la funcionalidad de Salmon que permite tener en cuenta el sesgo de gcs y para las lecturas single-end se utiliza el comando validateMappings
Tiempo de ejecución/muestra: 5 min.
El porcentaje de lecturas no mapeadas por muestra es cercano al 0% en todas ellas.

```
outdir="/home/karu/share/Salmon_quant/"
readsdir_paired="/home/karu/share/chr1/100_20X/together_paired_100_20X"
readsdir_single="/home/karu/share/chr1/100_20X/together_single_100_20X"
tail=".fasta" # part of the fastq filename after the sample name
transcripts_index="/home/karu/share/Definitivo/transcripts_index_salmon"
first='_1'
second='_2'
ths=6 #6
paired=$1 #1 = Paired #0 False
echo ""
sgelogsdir=$outdir/sgelogsdir
mkdir -p $sgelogsdir
#"-hold_jid 26886"

for lsname in `ls $readsdir_paired | grep $tail | cut -f1,2 -d'_' | uniq`
do
    if (echo "${paired}" | grep "1")
    then
#-l IU Inward unstranded como dice polyester
    salmon quant -i $transcripts_index -l IU --gcBias -1 $readsdir_paired/$lsname$first$tail -2 $readsdir_paired/$lsname$second$tail
        --validateMappings -o $outdir"paired_"$lsname
    else
    echo 'single'
    salmon quant -i $transcripts_index -l IU --gcBias -r $readsdir_single/$lsname$tail --validateMappings -o $outdir"single_"$lsname
    fi
done
```

4.3 Cuantificación y uso diferencial de transcrito

4.3.1 Salmón

En cuanto a la cuantificación, Salmon directamente nos devuelve la medida de transcritos por millón (TPMs) para cada transcrito, con un archivo para cada replicado.

4.3.1.1 DrimSeq

Para el análisis de los datos con DrimSeq partiendo de ASApp puede hacerse uso de la herramienta de R *tximport*, que facilita la importación de los datos, su anotación y su adaptación a DrimSeq, que directamente es capaz de recibirlos como input directamente. Es importante anotar posteriormente este objeto que nos devuelve tximport, para obtener la equivalencia de genes a los que pertenece cada transcrito, ya que DrimSeq necesita esta información para poder calcular las proporciones de cada isoforma respecto al total de transcritos que forman parte de ese grupo. Una vez obtenemos este objeto, podremos utilizar directamente DrimSeq, ya que es un paquete de R. Antes de ajustar el modelo, es útil realizar un proceso de filtrado, que acelerará el tiempo de ejecución de DrimSeq y evitará que cometa algunos errores cuando el número total de cuentas de un gen es muy bajo. Para ello, podemos filtrar por los transcritos que no tengan al menos 10 cuentas para al menos 6 réplicas, si la abundancia relativa del transcrito no supera cierta proporción o si el total de cuentas del gen es pequeño. Estos valores deberán adaptarse en función del tamaño muestral.

```
data_single <- dmFilter(data_single,
                        min_samps_feature_expr=n.small, min_feature_expr=10,
                        min_samps_feature_prop=n.small, min_feature_prop=0.01,
                        min_samps_gene_expr=n_samples, min_gene_expr=10)
data_single
```

Esto son los resultados de antes del filtrado y después del mismo:

Número de genes

Antes del filtrado	Después del filtrado
5705	1820

El filtrado se realizó con condiciones menos exigentes que las del manual, debido a la reducida dimensionalidad de nuestro experimento

Posteriormente, ya podemos estimar los parámetros necesarios para el ajuste con la función *dmPrecision*, que por defecto se realiza con un 10% de los genes, ya que este valor solo se utilizará como inicial y para ahorrar tiempo de computación. Así obtendremos una estimación de la expresión media de los genes por condición, por lo que es necesario pasarle una matriz de diseño donde indiquemos a qué condición pertenece cada una de las muestras analizadas. Posteriormente ajustamos el modelo y obtenemos las proporciones de cada uno de los transcritos y su verosimilitud.

- Tiempo de ejecución: 7 minutos

Debido a que DRIMSeq nos devuelve directamente todas las proporciones de todos los transcritos que ha detectado en la muestra, realizamos un filtrado de aquellos en los que hay un cambio de isoforma con un FDR para el cambio de proporción de isoforma < 0.05. Estos datos son los que utilizaremos para comparar con el resto de los pipelines. Además, DRIMSeq permite obtener por defecto algunas gráficas de la distribución de los datos. Además, se puede obtener gráficas sobre el cambio de isoformas entre condiciones para cada uno de los genes, permitiendo, tomando el gen con menor FDR, ver la diferencia entre condiciones que present (Figura 33). Por último, obteniendo la tabla que presenta el porcentaje de isoformas por condición, teniendo anotado los transcritos a los que pertenece, podremos escribir una tabla que nos permitirá realizar las comparaciones entre métodos.

4.3.1.2 ASapp

Como está especificado en la guía de ASapp, necesita un input específico donde se especifiquen la id de los transcritos, la id de los genes y el porcentaje de isoforma (IsoPct). Para detectar cambios, se ajusta para que la probabilidad de que pertenezca a la clase sufre switching o no sufre switching tenga una probabilidad mayor al 95%.

- Para calcular los IsoPcts, necesitamos anotar la matriz de Salmon con los genes a los que pertenece cada transcrito, haciendo uso del paquete tximport[43]. Posteriormente, es necesario que computemos estos valores para cada una de las muestras, por lo que mediante un bucle y el uso de la función *aggregate*, podemos obtener los TPMs totales para los transcritos que pertenecen a los mismos genes. Una vez hemos conseguido este valor, juntamos todas las muestras en una única matriz

- Mediante la unión de esta matriz de expresión total para cada gen, y juntándolo con una matriz de expresión en TPMs para cada transcrito utilizando la unión por el id del gen, podremos dividir los valores de ambas columnas y multiplicarlo por 100, obteniendo los TPMs (Figura 34).

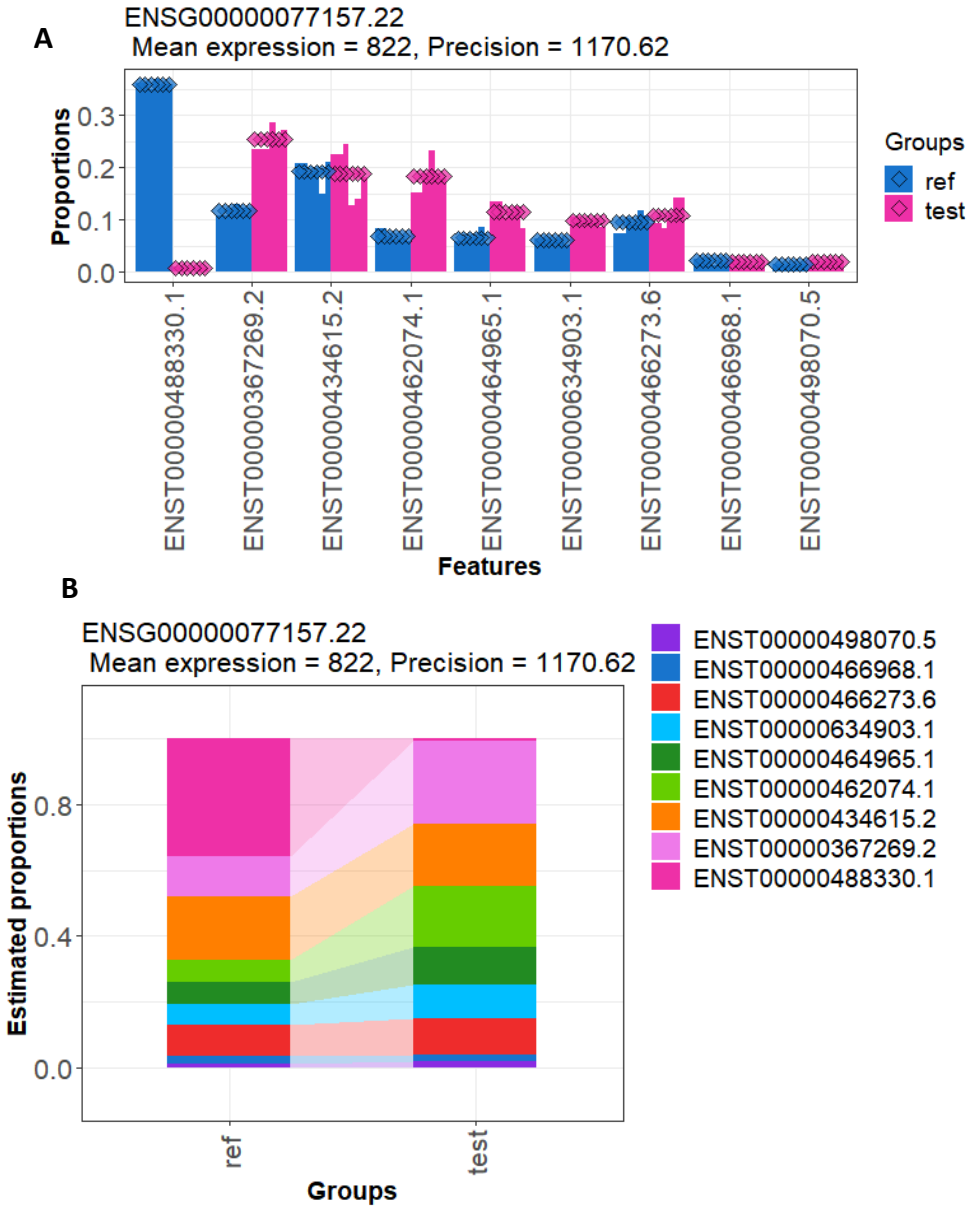


Figura 33. Gráficas de proporción de isoforma por condición para el gen que presenta mayor expresión diferencial. En la figura (**a**) podemos ver una comparación directa entre la proporción de cada isoforma en el gen. Se observa claramente un switch de la isoforma mayoritaria ENST00000488330.1, que casi desaparece por completo, y que se ve compensado por el aumento en otras dos isoformas. Estudiar estos cambios y la función de cada una de las formas implicadas en él nos podría dar información funcional.

Tras la obtención automática de los archivos en el formato y extensión adecuados, directamente podemos introducirlos de manera manual por la interfaz de ASapp. El tiempo de ejecución de ASapp con estos datos es de 50 segundos. En la salida, ASapp nos aporta varia información importante

directamente, sin necesidad de invocar la creación de ninguna gráfica a partir de los datos: nos muestra, tanto para la referencia como para la condición de test, las isoformas que han sufrido *switching* y si esa isoforma es la isoforma mayoritaria o no en términos de abundancia en esa condición (Figura 34).

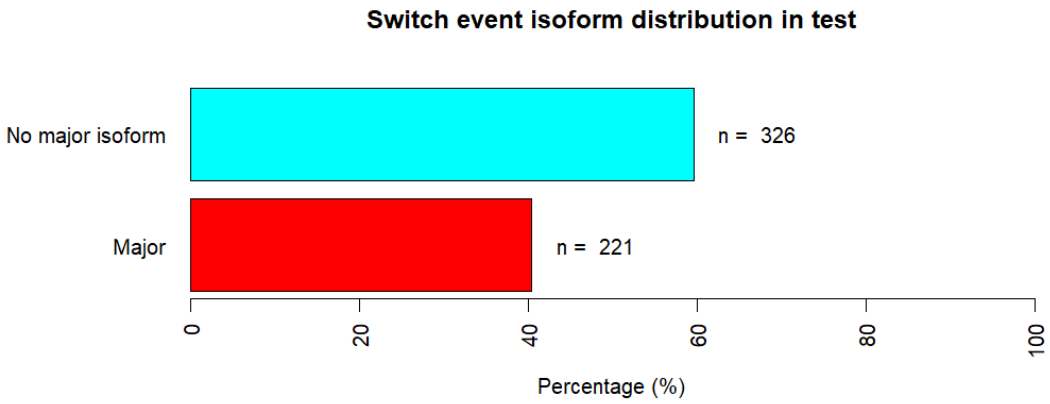


Figura 34. Representación de las isoformas afectadas por switching.

Además, podemos obtener la tabla que nos presenta directamente el porcentaje de isoformas para el caso problema y test, permitiéndonos hacer comparaciones entre los pipelines.

Como la salida de Salmón es simplemente los TPMs y ASapp no tiene un paquete que ayude a su adaptación, se desarrolló un script que permitirá la transformación para permitir su incorporación a ASapp vía interfaz gráfica. Partiendo del uso del paquete tximport, importamos directamente las lecturas de salmon, anotándolas con el objetivo de poder determinar el porcentaje de isoformas para cada uno de los transcritos, motivo por el cuál necesitamos saber de qué gen proviene cada isoforma y sumar el total de las isoformas que pertenecen al mismo, para posteriormente dividir cada una de ellas por el total y multiplicando por 100, obteniéndose el porcentaje(Figura 35).

```
for (sample_id in information$sample_id) {
  temporal<- aggregate(counts_single[[sample_id]], by=list(gene_id=counts_single$gene_id), FUN=sum)
  names(temporal) <- c('gene_id', eval(parse(text = "sample_id")))
  assign(eval(parse(text = "sample_id")), temporal)
}
merge_single <- Reduce(function(x, y) merge(x, y, all=TRUE), list(sample_01,sample_02,sample_03,sample_04,sample_05,sample_06,
                                                                  sample_07,sample_08,sample_09,sample_10,sample_11,sample_12))

names <- names(counts_single)
for (sample_id in information$sample_id) {
  sample_matrix <- as.data.frame(cbind(counts_single$gene_id, counts_single$feature_id,counts_single[[sample_id]]))
  names(sample_matrix) <- c("gene_id", "transcript_id", sample_id)
  sample_matrix[[sample_id]] <- as.numeric(as.character(sample_matrix[[sample_id]]))
  temporal_merge <- as.data.frame(cbind(merge_single$gene_id, merge_single[[sample_id]]))
  names(temporal_merge) <- c('gene_id',sample_id)
  temporal_merge[[sample_id]] <- as.numeric(as.character(temporal_merge[[sample_id]]))
  sample_matrix <- merge(sample_matrix,temporal_merge, by='gene_id')
  sample_matrix <- transform(sample_matrix, IsoPct = (sample_matrix[,3]/sample_matrix[,4])*100)
  assign(eval(parse(text = "sample_id")), sample_matrix)
  write.table(sample_matrix, file = paste0('E:/Documentos/Master_Bioinformatica/TFM/Quant/Salmon_quant/Single/',
  sample_id,".isoforms.results"),row.names=FALSE, quote = FALSE, sep='\t')

}
```

Figura 35. Fragmento de la obtención del porcentaje de isoformas muestras a partir de la matriz que devuelve Salmon. Representa el bucle donde podemos obtener el denominador para realizar la normalización. Lo realiza para cada una de las muestras de manera automática, en función de la matriz de diseño que le aportes al inicio.

4.3.1.3 SUPPA

Al ser un programa inicialmente pensado para detectar los eventos de splicing alternativo, la adición de esta nueva funcionalidad es interesante de llevarse a estudio. Para ello, directamente presenta adaptaciones de la salida de Salmon mediante el uso secuencial de varios *scripts* y los comandos *generateEvents* y *psiPerIsoform*. El primero nos genera un archive ioi a partir del archivo.gtf de la versión que hayamos utilizado en todos los pasos anteriores: aquí guarda la información de, para cada transcrito del gen, todos los transcritos del mismo. El segundo nos permite calcular el valor de psi para cada isoforma, que actuaría en lugar del evento. Posteriormente dividimos el archivo en función de las muestras que pertenecen a cada condición y directamente podemos calcular la expresión diferencial con la función de suppa.py *diffSplice.* El tiempo de ejecución fueron 9 minutos. Para la comparación, seleccionaremos como isoformas que presentan switching aquellas que presenten un FDR < 0.05.

4.3.2 RSEM

Para favorecer la continuidad del pipeline, se generó un script que permite la adaptación de la salida de STAR para su ejecución automática en RSEM para cada muestra, y que permite pasar por la línea de comando todas las opciones oportunas, como los directorios de entrada y salida, si las lecturas son paired o single, el nombre de las referencias, etc, y que extraerá directamente los nombres de los outputs y los generará en carpetas diferentes en función de todas las muestras de STAR (.bam) que se encuentren en el directorio seleccionado. Se realizó una versión que permitía la adaptación para enviar los trabajos de forma masiva al clúster. El tiempo de ejecución aproximada por muestra es de 6 minutos.

4.3.2.1 DrimSeq

Se sigue el mismo pipeline que en el caso anterior, ya que la herramienta tximport también permite importar directamente los datos de rsem como un objeto txdb. Se observa que el número de genes tras el filtrado es algo mayor con RSEM que en el caso de Salmon.

Número de genes

Antes del filtrado	Después del filtrado
5705	2219

El tiempo que tarda en correr todo el proceso de ajuste y cálculos de DrimSeq con la salida de RSEM es de aproximadamente 7.4 minutos.

4.3.2.2 ASapp

La herramienta ASapp acepta directamente las cuantificaciones realizadas por RSEM, ya que presentan el porcentaje de isoforma. Además, el formato que devuelve RSEM es compatible con el input exigido por el programa. Al observar el gráfico de distribución de isoformas expresadas diferencialmente y si esta es la isoforma mayoritaria o no, se ve que claramente con RSEM tenemos una mayor detección de *switching* respecto a la condición de Salmon (Figura 36). Cabe destacar el incremento de tiempo de computación al utilizar directamente el output de RSEM respecto a la selección más moderada de columnas que hacíamos al adaptar la salida de Salmón (20 min vs 50 segundos). Se podría seleccionar previamente las únicas tres columnas necesarias de RSEM para reducir este tiempo.

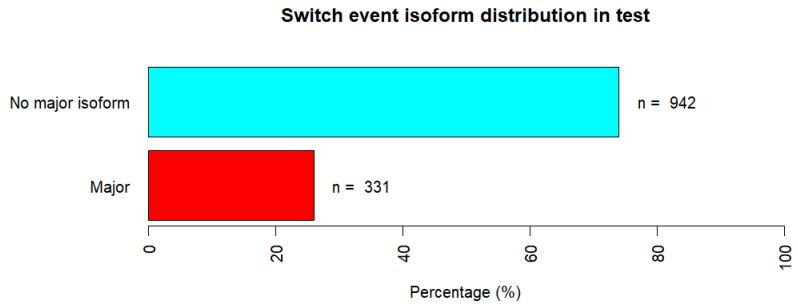


Figura 36. Visualización de los eventos de switching detectados por ASapp cuando se ejecuta en el pipeline STAR+RSEM+ASapp

4.4 Análisis de precisión de los pipelines

Para poder comparar todos los pipelines necesitamos una medida sobre la cual realizar todos los cálculos. Para ello, con el uso de las cuentas por isoforma que obtenemos como realidad a partir de *Polyester* para poder realizar las estimaciones, se adaptará un script que permitirá el cálculo de los TPMs y el porcentaje de isoformas a partir de estas cuentas, ya que la comparación se hará a nivel de cambios en el porcentaje. Una vez tenemos estos resultados, calculamos el $\log_2 Fold\ Change$, siendo este el porcentaje o proporción de esa isoforma en la referencia dividido entre el porcentaje o proporción de esa isoforma en la condición de test. Estos valores, obtenidos para nuestras cuentas simuladas, actuarán como el valor real sobre el cual calcularemos el error cuadrático medio (MSE) respecto a cada pipeline. Como se puede observar en la Figura 37, el pipeline que menos error presenta es el que cuantifica con RSEM y se analiza con ASapp. Y, sin embargo, la pipeline RSEM+DrimSeq apenas ha conseguido detectar bien las diferencias, ya que solo permitía la permanencia de 10 transcritos después de filtrar por el FDR<0.05.

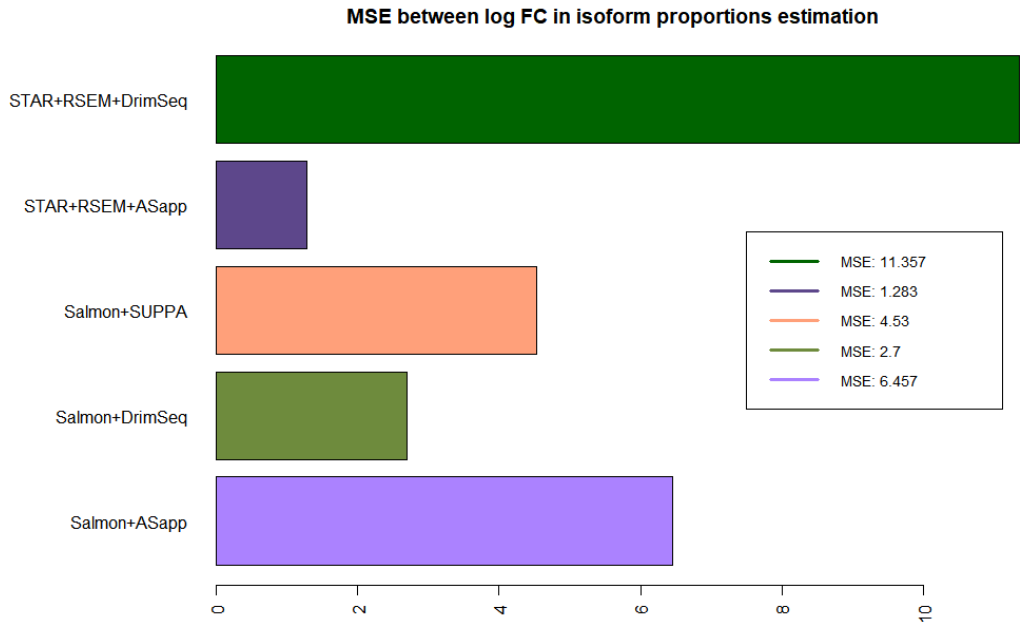


Figura 37. Cálculo del error cuadrático de la media para cada uno de los pipelines escogidos.

Para ver la capacidad de detección de cada uno de los métodos, se utilizó como punto de corte un log_2fold change > 1 o log_2fold change <-1, que serían cambios de 2 veces la expresión, tanto por arriba como por abajo. A esas isoformas se les puso la etiqueta 1 que simboliza que sufre switching, mientras que el resto de transcritos (la mayoría, ya que en *Polyester* decidimos guiar de esta manera la simulación para acercarse a lo que ocurre en el interior de las células) tendrían una etiqueta de 0 y no tendrían que ser detectados por nuestros programas como isoforma que sufre switching. De esta manera, y clasificando como sufre switching a todas las isoformas que se detectan en cada método y como 0 el resto, se elaboró una curva ROC para cada una de las pipelines (Figura 38). Podemos observar que las dos pipeline que presentan un mejor resultado son las que utilizan ASapp, siendo mejor el resultado entre estas dos cuando el cuantificador es RSEM.

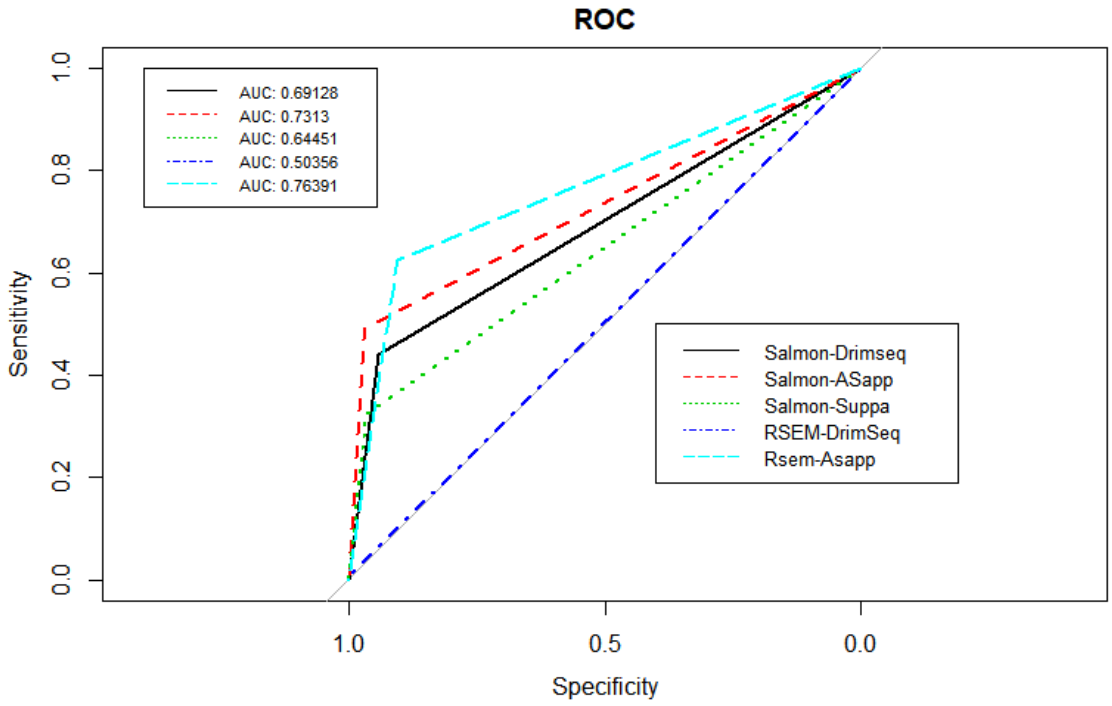


Figura 38. Curva ROC de Sensibilidad vs Especificidad para cada uno de los pipelines escogidos.

Por último, y con el objetivo de entender de mejor manera los resultados de la curva ROC y la clasificación que ha realizado cada clasificador, se realizó diferentes intersecciones entre los transcritos que se consideraba que sufrían switching entre distintas pipelines y se representó en un Diagrama de Venn. Debido al problema que presenta la pipeline de DrimSeq, la primera comparación que se realizó fue entre todas las posibilidades que tenía la pipeline de Salmon, tanto compara con entre las tres opciones (ASApp, DRIMSeq, SUPPA) Figura 39**A**, como viendo cuales de estas coincidían con las detectadas por el *ground truth* Figura 39**B**. Se puede ver como hay una mayor intersección entre ASapp y DRIMSeq a la hora de detectar estas diferencias, pero que no todas ellas son detectadas en el *ground truth* con un cambio de el doble en la proporción de las isoformas. También se puede observar como claramente DRIMSeq tiene una mayor capacidad de detección respecto a los otros pipelines, ya que presenta más isoformas exclusivas, pero se ve que mayoría de estas son falsos positivos (Figura 39**B**). En cuanto a la

sensibilidad, puede verse como ASApp presenta una mayor proporción de verdaderos positivos (333 verdaderos positivos respecto a 568 falsos positivos 0.58) respecto a DRIMSeq (275 verdaderos positivos frente a 1026) y SUPPA (195). Estos resultados concuerdan con los anteriormente descritos por la curva ROC.

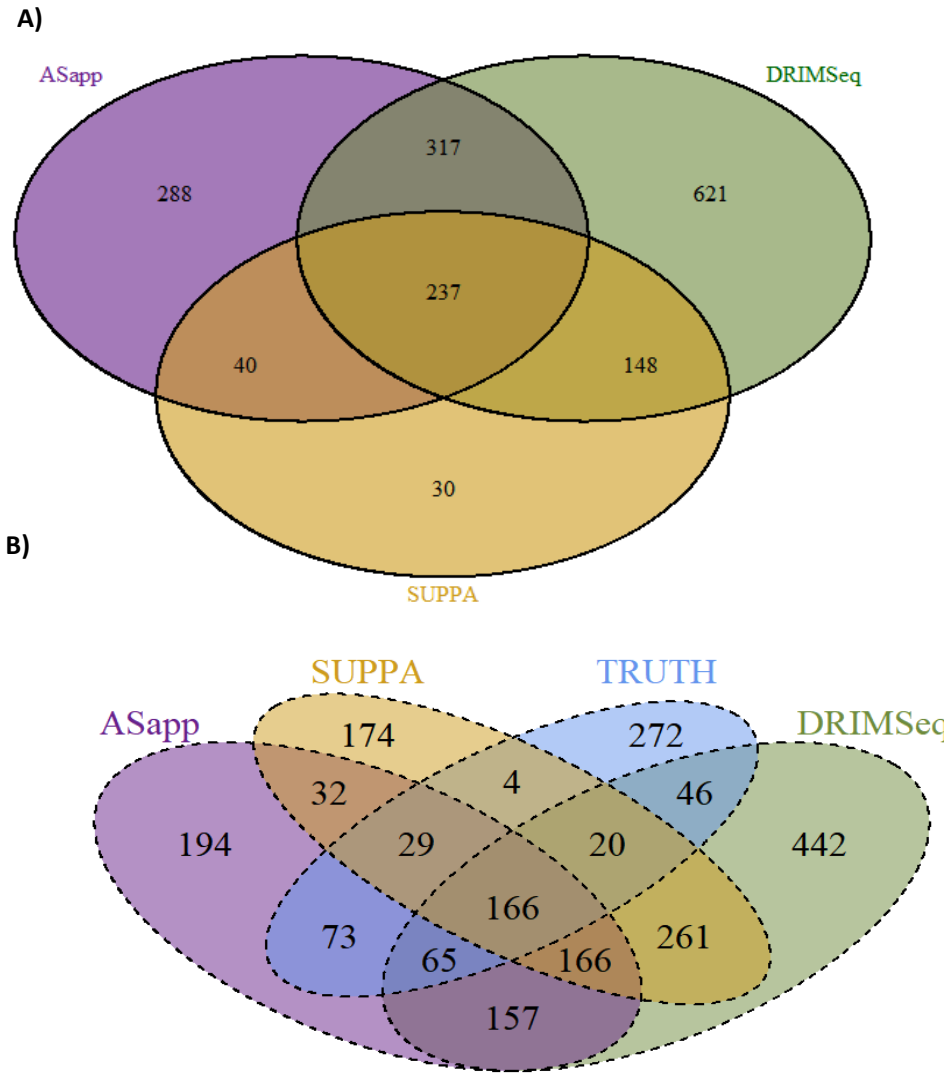


Figura 39. Diagrama de Venn entre las distintas opciones para detectar uso diferencial de transcrito en el pipeline de Salmon.

Respecto al pipeline de RSEM, al no encontrarse solapamientos entre los pocos transcritos detectados por DRIMSeq y los que detecta ASApp, se han realizado los diagramas de Venn para comparar, de cada uno, cuáles se corresponden con los verdaderos. Esto deja evidenciar que, por algún motivo de *bug* o incompatibilidad, no se ha podido detectar de forma correcta las isoformas obtenidas por Rsem-Drimseq. Además, el hecho de que la clasificación que realiza el pipeline RSEM+DrimSeq sea aproximadamente como el azar (Figura 38) nos hace excluirla de las comparativas al ser considerado

como un error. Se puede ver además que ASApp es capaz de detectar una mayor cantidad de isoformas cuando la cuantificación proviene de RSEM.

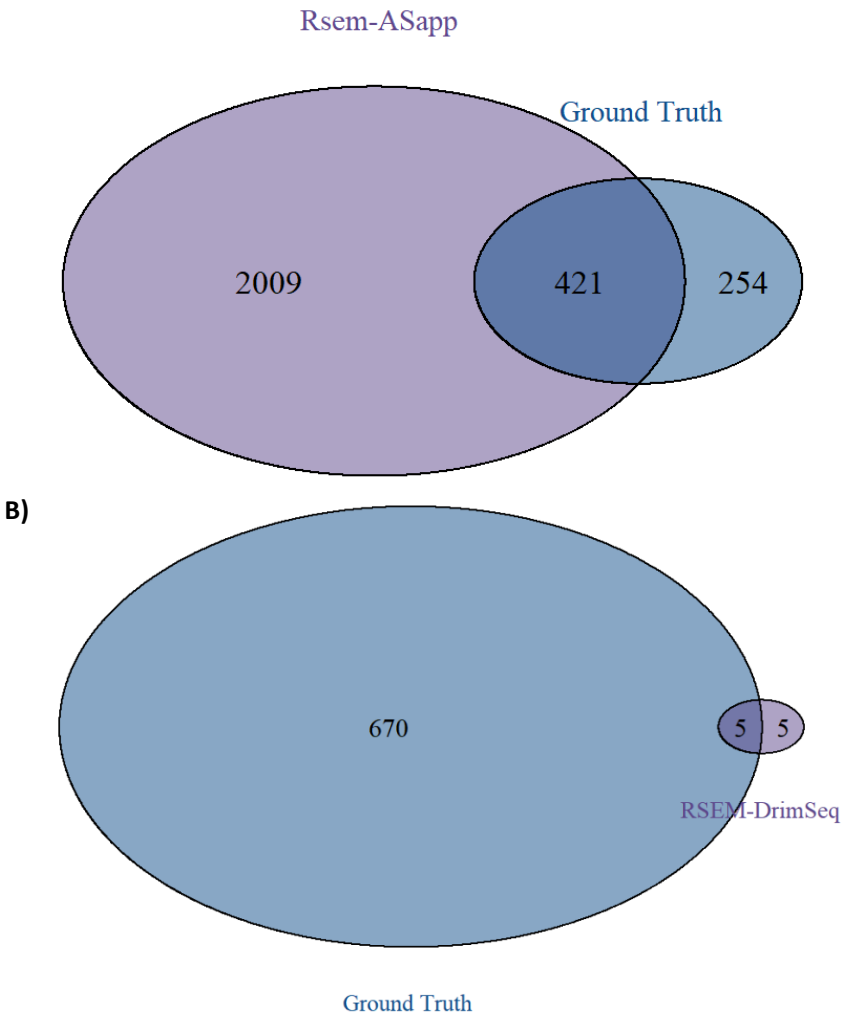


Figura 40. Diagrama de Venn que permite comparar, respecto a la "verdad", las isoformas detectadas que coinciden con las del pipeline indicada.

Por último, si comparamos los tiempos de ejecución entre las distintas condiciones (Figura 41), no se detectan grandes diferencias entre el método de detección de uso de isoforma, ya que no hay casi diferencia entre ellas cuando pertenecen al mismo clasificador. Además, el alto tiempo de RSEM+ASapp podría reducirse bastante mediante la implementación para obtener únicamente las tres columnas relevantes para el procesado. Entre las muestras de Salmon, podemos ver una cierta tendencia a la disminución del tiempo de ejecución, favoreciéndose el pipeline de Salmon y ASApp sobre todos los demás.

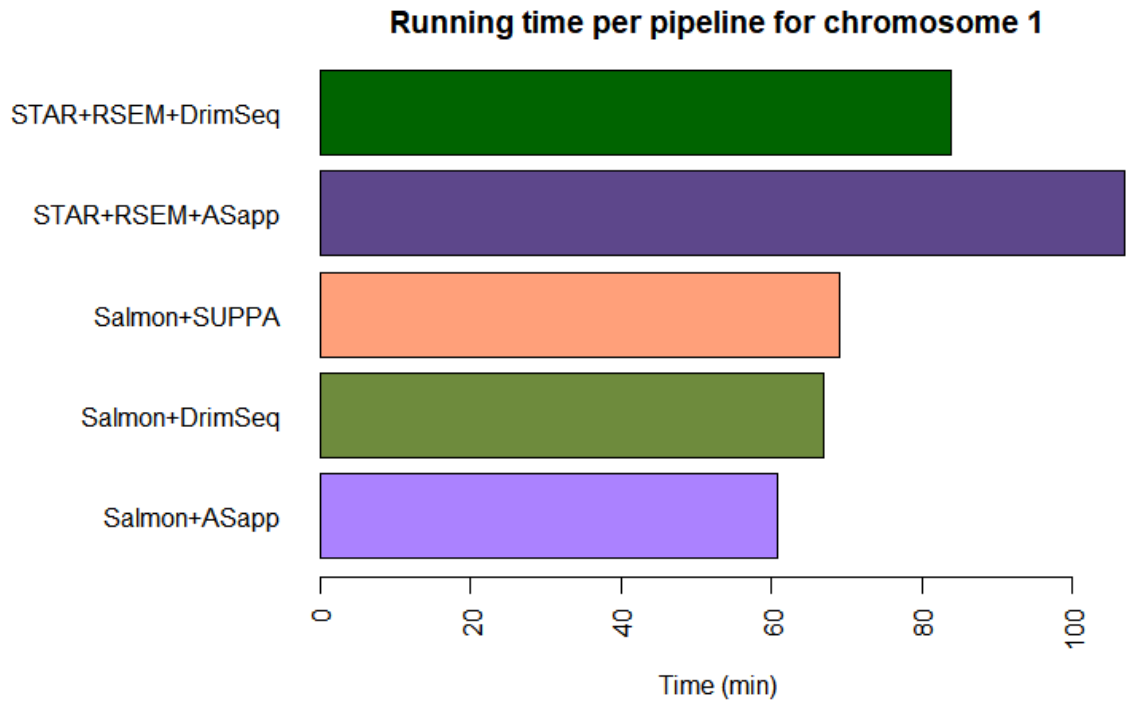


Figura 41. Tiempo para correr cada uno de los pipelines con el cromosoma. Es importante tener en cuenta que los tiempos reales con una muestra de transcriptómica serán mucho mayores, pero las comparaciones entre métodos siguen siendo válidas a pequeña escala

5 Conclusiones, expectativas y trabajo de futuro

Observando la curva ROC, las dos que presentan un mayor valor de AUC son RSEM+ASApp y Suppa+ASApp. Salmon+AsAPP es más específica, aunque presenta una menor sensibilidad (la curva ROC tiene una mayor pendiente al inicio, lo que quiere decir que aumenta mucho la sensibilidad sin necesidad de sacrificar la especificidad durante ese rango). Sin embargo, RSEM es capaz de alcanzar una mayor área bajo la curva ya que consigue incrementar mucho la sensibilidad a costa de sacrificar la especificidad, como se puede ver en la cantidad de falsos positivos que se aprencian en la Figura 40**A**.

Se ve que gracias a la capacidad de detección que presenta, ASApp es una buena herramienta para realizar la detección de uso diferencial de transcritos. La facilidad de uso y manejo la hacen una buena candidata para utilizarla para realizar análisis iniciales y rápidos, para luego seguir utilizando sus datos para calcular distintos parámetros que puedan resultar interesantes, como puede verse en la curva ROC (Figura 38) y en el diagrama de *Venn* de los pipelines de Salmon (Figura 39).

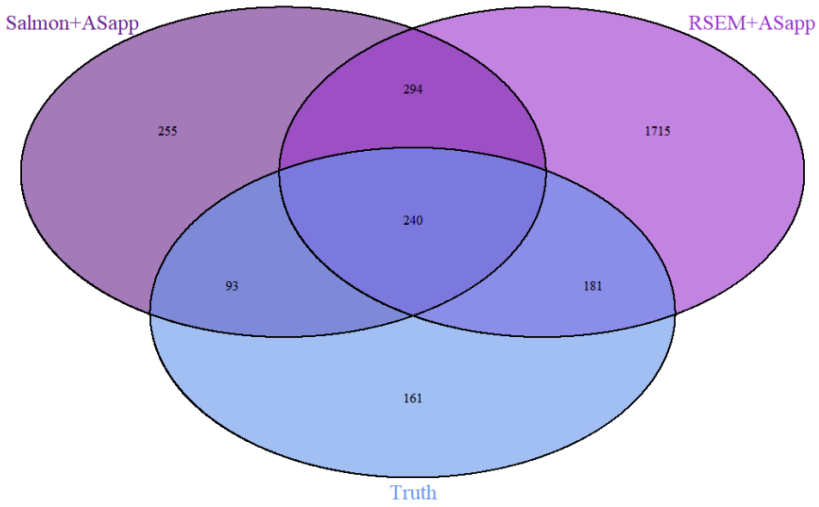


Figura 42. Comparación de los niveles de detección y de falsos positivos para ASApp en ambos pipelines.

Respecto al intercambio calidad/tiempo de ejecución y facilidades, sería difícil con los estudios iniciales establecer cuál de los pipelines sería el más adecuado a tener en cuenta, si RSEM+ASApp o Salmon+ASApp, debido a la rapidez y ventajas que presenta el segundo al no requerir de tanta capacidad computacional, pero la gran versatilidad y capacidad de detección de RSEM y su output preparado para ASApp lo hacen un buen candidato. Dependiendo de la necesidad del proyecto y las condiciones que se definan, podría ser más interesante utilizar una u otra: teniendo en cuenta que en estos experimentos siempre se esperarán algunos cambios en los transcritos y que no todos pueden ser estudiados posteriormente, quizá sería interesante realizar primero el procedimiento con Salmon, ya que es más rápido e introduce menor cantidad de falsos positivos, evitando que nos centremos en un transcrito que no presentaba diferencia real. Otra opción podría ser calcular la intersección de transcritos diferencialmente expresados entre distintos pipelines, ya que esos presentan un claro cambio para todos y pueden ser más adecuados para su estudio posterior.

Aunque el error en la estimación de la proporción de isoformas de Salmon-DRIMSeq respecto a Salmon-ASApp es algo menor (y al no poder compararse la versión de RSEM por la ausencia de resultados en DRIMSeq), ASApp permite detectar con mejor eficiencia los transcritos que presentan una diferencia en la isoforma, lo cual suele ser la aproximación deseada en estos estudios de transcriptómica. Además, el método que menor error presenta de manera global es RSEM+ASApp, y al no haber diferencias muy relevantes en los tiempos de ejecución si se realizan las modificaciones propuestas, con los datos actuales se puede concluir que ASApp es el método que permite detectar de forma correcta mayor número de isoformas que presentan uso diferencial de transcrito y que entre estos dos, Salmon-ASApp presenta una mayor especificidad para las sensibilidades más bajas, mientras que RSEM-Asapp es capaz de determinar un mayor número de isoformas mediante un sacrificio en esta especificidad. Además, el hecho de que la curva ROC de RSEM-ASapp sea la mejor a pesar de la inclusión de tantos falsos positivos respecto a Salmon-AsAPP (33% de las isoformas detectadas como positivas por Salmon ASApp lo son también en el ground truth respecto al al 17% de RSEM-Asapp se debe a que la gran inclusión de falsos positivos queda diluida entre la gran cantidad de verdaderos negativos presentes, debido al sesgo biológico de que el *switching* no es un evento muy común, y que RSEM-ASapp es capaz de captarlo. Si tenemos en cuenta que la metodología RSEM+ASApp es la más

adecuada en la predicción de los valores de isoformas reales pero que presenta una gran cantidad de falsos positivos, incrementando la restricción de ASApp para la asignación de verdaderos positivos a un 99% en este pipeline podría mejorar mucho los resultados.

En el futuro, la idea es ser capaz de, teniendo todos los pipelines estandarizados y con *scripts* para favorecer la adaptación entre cada uno de los pasos, poder generar muchas más condiciones, para ser capaz de ver como funciona cada uno de los métodos en función de las situaciones, y así poder establecer un criterio en su selección en función de los datos de partida que tengamos y de las posibilidades de tener distintas profundidades, longitudes, réplicas…

Además, se tiene pensado introducir, de manera manual, algunos *outliers* para comprobar la robustez del sistema en estas situaciones, mediante la selección de un 10% de los datos para enviarlos hacia los extremos y ver cual de los pipelines es capaz de soportar mejor esta desviación.

El objetivo final es ser capaz de crear un repositorio de github donde se puedan introducir todos los *scripts* de adaptación de los distintos pipelines, así como todas las automatizaciones en los procesos, para favorecer el análisis de cualquiera de los pipelines utilizados.

De manera más específica, se quiere poder observar como el número de exones que tiene el gen puede afectar o no a la capacidad de detección del uso de isoformas, y observar el comportamiento de cada método en función de este valor.

También se quiere desarrollar algunos scripts que permitan el análisis funcional posterior de cada uno de los métodos. Los valores que estamos midiendo aquí, incluso en el peor de los casos, son suficientes. Tenemos que tener en cuenta que abarcar 200 transcritos para estudiar la funcionalidad ya es bastante, por lo que tener mucha mayor cantidad tampoco sería beneficioso. En ese sentido, se está pensando en generar unos pequeños programas que, escogiendo las isoformas que más claramente presentan uso diferencial de transcrito, ser capaces de, anotándola de diferentes maneras, comprender si la nueva isoforma ha incorporado algún exón que represente un cambio importante por la introducción de un nuevo dominio funcional, entender la funcionalidad del gen del que provienen las isoformas e incluso ser capaz de filtrar por aquellos casos en los que una isoforma que es mayoritaria en una condición, se convierta en minoritaria en la otra. Se busca también realizar algunas mejoras en la implementación de ASApp, solución de errores… Entre las mejoras propuestas, está el hecho de incorporar a ASApp, debido a que ya es bastante eficiente y rápida y no incrementaría mucho el tiempo de computación, alguna herramienta que permita una mejor anotación funcional de los transcritos y así poder ver si el *switching* implica grandes cambios a nivel de dominios (Pfam), pequeñas regiones… e intentar predecir un posible efecto en la función. Por ejemplo: generalmente, cuando en el splicing alternativo se produce retención de un intrón esto suele llevar a que este ARN se mantenga en el núcleo o sufra una degradación debido a la aparición de una mutación terminadora (NMD o *nonsense-mediated decay*). Sin embargo, si ese switching que hemos observado implicase una retención de intrón y aun así mantiene la pauta de lectura, se ha asociado este cambio con determinadas funciones importantes[32]. También se propuso la incorporación de la base de datos ATtRACT para ASApp, para intentar ver si la isoformas afectadas por el *switching* ha supuesto la incorporación de alguna región que posea sitios de unión a proteínas de unión a ARN o RBPs[62]. La adición de estos análisis funcionales podría darle un nuevo camino al uso diferencial de tránscritos, permitiendo detectar realmente qué parte de la variabilidad biológica puede explicarse por este fenómeno y poner fin a la controversia.

Glosario de acrónimos (orden alfabético)

- **ALD:** Asignación Latente de Dirichlet
- **ADNc**: ADN copia
- **CAGE**: Análisis del cap de la expresión génica (Cap Analysis of Gene Expression)
- **EM:** Esperanza-Maximización (*Expectation-Maximization*)
- **EST**: Expressed Sequence Tag o marcador de secuencia expresada.
- **JSD:** Divergencia de Jensen-Shannon (*Jensen-Shannon Divergency*)
- **ARNm:** ARN mensajero.
- **MEM:** Secuencias exactas máximas (*Maximal Exact Matches*)
- **MMP:** Prefijo máximo mapeable (*Maximal Mapeable Prefix*)
- **MSE:** Error cuadrático medio (*Mean squared error*)
- **NGS**: Secuenciación de nueva generación (Next generation sequencing)
- **PCR**: Reacción en cadena de polimerasa o Polymerase Chain Reaction
- **PTBP:** Proteína de unión al tracto de polipirimidinas (Polypyrimiding tract binding protein)
- **RNA-seq**: Secuenciación masiva del ARN
- **RNPs:** Proteínas de unión a ARN (RNA binding protein)
- **RSEM:** ARN-seq por Esperanza-Maximización (*RNA-Seq Expectation-Maximization*)
- **SAGE**: Análisis en Serie de la Expresión Génica (Serial Analysis of Gene Expression)
- **SMEMs:** Secuencias super exactas máximas (*Super maximal exact matches)*
- **snARNs:** ARNs nucleares pequeños (small nuclear RNAs)
- **snRNP:** Ribonucleoproteínas nucleares pequeñas (small nuclear ribonucleoproteins)
- **SREs:** Elementos reguladores de splicing (Splicing regulatory element)
- **TPMs:** Transcritos por millón
- **URE:** Elemento regulador anterior (Upstream regulatory element)

Bibliografía

[1] J. Huang *et al.*, "Coding and noncoding gene expression biomarkers in mood disorders and schizophrenia.," *Dis. Markers*, vol. 35, no. 1, pp. 11–21, 2013.

[2] M. Irimia *et al.*, "A highly conserved program of neuronal microexons is misregulated in autistic brains," *Cell*, vol. 159, no. 7, pp. 1511–1523, 2014.

[3] C. Trapnell *et al.*, "Transcript assembly and abundance estimation from RNA-Seq reveals thousands of new transcripts and switching among isoforms," *Nat. Biotechnol.*, vol. 28, no. 5, pp. 511–515, 2011.

[4] R. Patro, G. Duggal, M. I. Love, R. A. Irizarry, and C. Kingsford, "Salmon provides fast and bias-aware quantification of transcript expression," *Nat. Methods*, vol. 14, no. 4, pp. 417–419, Apr. 2017.

[5] B. Li and C. N. Dewey, "RSEM: Accurate transcript quantification from RNA-seq data with or without a reference genome," *Bioinforma. Impact Accurate Quantif. Proteomic Genet. Anal. Res.*, pp. 41–74, 2014.

[6] M. Nowicka and M. D. Robinson, "DRIMSeq: a Dirichlet-multinomial framework for multivariate count outcomes in genomics," *F1000Research*, vol. 5, no. 0, p. 1356, 2016.

[7] J. L. Trincado *et al.*, "SUPPA2: Fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions," *Genome Biol.*, vol. 19, no. 1, pp. 1–11, 2018.

[8] A. C. Frazee, A. E. Jaffe, B. Langmead, and J. T. Leek, "Polyester: Simulating RNA-seq datasets with differential transcript expression," *Bioinformatics*, vol. 31, no. 17, pp. 2778–2784, 2015.

[9] Y. Benjamini and T. P. Speed, "Summarizing and correcting the GC content bias in high-throughput sequencing," *Nucleic Acids Res.*, vol. 40, no. 10, pp. 1–14, 2012.

[10] ENCODE, "Current ENCODE Experiement Guidelines," no. December, p. 5, 2017.

[11] S. Roychowdhury and A. M. Chinnaiyan, "Translating cancer genomes and transcriptomes for precision oncology," *CA. Cancer J. Clin.*, vol. 66, no. 1, pp. 75–88, 2015.

[12] M. Alshalalfa, M. Schliekelman, H. Shin, N. Erho, and E. Davicioni, "Evolving transcriptomic fingerprint based on genome-wide data as prognostic tools in prostate cancer," *Biol. Cell*, vol. 107, no. 7, pp. 232–244, 2015.

[13] M. Alvarez, A. W. Schrey, and C. L. Richards, "Ten years of transcriptomics in wild populations: What have we learned about their ecology and evolution?," *Mol. Ecol.*, vol. 24, no. 4, pp. 710–725, 2015.

[14] N. Shirley, T. Shafee, S. Dolan, R. Lowe, and M. Bleackley, "Transcriptomics technologies," *PLOS Comput. Biol.*, vol. 13, no. 5, p. e1005457, 2017.

[15] M. Adams *et al.*, "Complementary DNA sequencing: expressed sequence tags and human genome project," *Science (80-.).*, vol. 252, no. 5013, pp. 1651–1656, 1991.

[16] M. Blaxter and J. Parkinson, “Expressed Sequence Tags: An Overview,” *Methods Mol. Biol. B. Ser.*, vol. 533, pp. 1–12, 2009.
[17] S. H. Nagaraj, R. B. Gasser, and S. Ranganathan, “A hitchhiker’s guide to expressed sequence tag (EST) analysis,” *Brief. Bioinform.*, vol. 8, no. 1, pp. 6–21, 2007.
[18] R. Alba *et al.*, “ESTs, cDNA microarrays, and gene expression profiling: Tools for dissecting plant physiology and development,” *Plant J.*, vol. 39, no. 5, pp. 697–714, 2004.
[19] Z. Fei *et al.*, “Comprehensive EST analysis of tomato and comparative genomics of fruit ripening,” *Plant J.*, vol. 40, no. 1, pp. 47–59, 2004.
[20] M. Sun, G. Zhou, S. Lee, J. Chen, R. Z. Shi, and S. M. Wang, “SAGE is far more sensitive than EST for detecting low-abundance transcripts.,” *BMC Genomics*, vol. 5, no. 1, p. 1, Jan. 2004.
[21] Z. Wang, M. Gerstein, and M. Snyder, “RNA-Seq: a revolutionary tool for transcriptomics,” *Nat. Rev. Genet.*, vol. 10, no. 1, pp. 57–63, Jan. 2009.
[22] M. F. Rai, E. D. Tycksen, L. J. Sandell, and R. H. Brophy, “Advantages of RNA-seq compared to RNA microarrays for transcriptome profiling of anterior cruciate ligament tears,” *J. Orthop. Res.*, vol. 36, no. 1, pp. 484–497, 2018.
[23] J. Wang, D. C. Dean, F. J. Hornicek, H. Shi, and Z. Duan, “RNA sequencing (RNA-Seq) and its application in ovarian cancer,” *Gynecol. Oncol.*, vol. 152, no. 1, pp. 194–201, 2019.
[24] Simon Andrews, “FastQC A Quality Control tool for High Throughput Sequence Data,” *Babraham Bioinformatics*, 2010. [Online]. Available: https://www.bioinformatics.babraham.ac.uk/projects/fastqc/. [Accessed: 06-Jun-2019].
[25] C. Soneson, K. L. Matthes, M. Nowicka, C. W. Law, and M. D. Robinson, “Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage,” *Genome Biol.*, vol. 17, no. 1, pp. 1–15, 2016.
[26] V. La Cognata *et al.*, “Increasing the Coding Potential of Genomes Through Alternative Splicing: The Case of PARK2 Gene,” *Curr. Genomics*, vol. 15, no. 3, p. 203, Jun. 2014.
[27] Y. Wang and Z. Wang, “Systematical identification of splicing regulatory cis-elements and cognate trans-factors,” *Methods*, vol. 65, no. 3, pp. 350–358, 2014.
[28] M. P. Mullen, C. W. Smith, J. G. Patton, and B. Nadal-Ginard, “Alpha-tropomyosin mutually exclusive exon selection: competition between branchpoint/polypyrimidine tracts determines default exon choice.,” *Genes Dev.*, vol. 5, no. 4, pp. 642–55, Apr. 1991.
[29] M. C. Wollerton, C. Gooding, F. Robinson, E. C. Brown, R. J. Jackson, and C. W. J. Smith, “Differential alternative splicing activity of isoforms of polypyrimidine tract binding protein (PTB),” 2001.
[30] M. L. Tress, F. Abascal, and A. Valencia, “Most Alternative Isoforms Are Not Functionally Important,” *Trends Biochem. Sci.*, vol. 42, no. 6, pp. 408–410, 2017.
[31] B. J. Blencowe, “The Relationship between Alternative Splicing and Proteomic

Complexity," *Trends Biochem. Sci.*, vol. 42, no. 6, pp. 407–408, 2017.
[32] R. J. Weatheritt, T. Sterne-Weiler, and B. J. Blencowe, "The ribosome-engaged landscape of alternative splicing," *Nat. Struct. Mol. Biol.*, vol. 23, no. 12, pp. 1117–1123, 2016.
[33] M. L. Tress, F. Abascal, and A. Valencia, "Alternative Splicing May Not Be the Key to Proteome Complexity," *Trends Biochem. Sci.*, vol. 42, no. 2, pp. 98–110, 2017.
[34] I. Ezkurdia, J. M. Rodriguez, E. Carrillo-De Santa Pau, J. Vázquez, A. Valencia, and M. L. Tress, "Most highly expressed protein-coding genes have a single dominant isoform," *J. Proteome Res.*, vol. 14, no. 4, pp. 1880–1887, 2015.
[35] M. L. Tress *et al.*, "The implications of alternative splicing in the ENCODE protein complement," *Proc. Natl. Acad. Sci.*, vol. 104, no. 13, pp. 5495–5500, 2007.
[36] M. Gstaiger and R. Aebersold, "Applying mass spectrometry-based proteomics to genetics, genomics and network biology," *Nat. Rev. Genet.*, vol. 10, no. 9, pp. 617–627, 2009.
[37] F. Abascal *et al.*, "Alternatively Spliced Homologous Exons Have Ancient Origins and Are Highly Expressed at the Protein Level," *PLoS Comput. Biol.*, vol. 11, no. 6, pp. 1–29, 2015.
[38] J. M. Mudge *et al.*, "The origins, evolution, and functional potential of alternative splicing in vertebrates," *Mol. Biol. Evol.*, vol. 28, no. 10, pp. 2949–2959, 2011.
[39] Y. Xia *et al.*, "Widespread Expansion of Protein Interaction Capabilities by Alternative Splicing," *Cell*, vol. 164, no. 4, pp. 805–817, 2016.
[40] J. Valcárcel *et al.*, "Large-scale analysis of genome and transcriptome alterations in multiple tumors unveils novel cancer-relevant splicing networks," *Genome Res.*, vol. 28, no. 9, pp. 1426–1426, 2018.
[41] P. I. Poulikakos *et al.*, "RAF inhibitor resistance is mediated by dimerization of aberrantly spliced BRAF(V600E)," *Nature*, vol. 480, no. 7377, pp. 387–390, Dec. 2011.
[42] E. Sebestyén, M. Zawisza, and E. Eyras, "Detection of recurrent alternative splicing switches in tumor samples reveals novel signatures of cancer," *Nucleic Acids Res.*, vol. 43, no. 3, pp. 1345–1356, 2015.
[43] C. Soneson, M. I. Love, and M. D. Robinson, "Differential analyses for RNA-seq : transcript-level estimates improve gene-level inferences [version 1 ; referees : 2 approved]," *F1000Research*, no. 4, pp. 1521–15, 2015.
[44] S. Anders and W. Huber, "Differential expression analysis for sequence count data," *Genome Biol.*, vol. 11, no. 10, p. R106, Oct. 2010.
[45] J. Lipp, "Why sequencing data is modeled as negative binomial," 2016. [Online]. Available: https://bioramble.wordpress.com/2016/01/30/why-sequencing-data-is-modeled-as-negative-binomial/. [Accessed: 12-Jun-2019].
[46] M. Garber, M. G. Grabherr, M. Guttman, and C. Trapnell, "Computational methods for transcriptome annotation and quantification using RNA-seq," *Nat. Methods*, vol. 8, no. 6, pp. 469–477, 2011.
[47] A. Dobin *et al.*, "STAR: Ultrafast universal RNA-seq aligner," *Bioinformatics*, vol. 29,

no. 1, pp. 15–21, 2013.
[48] A. Srivastava, H. Sarkar, N. Gupta, and R. Patro, “RapMap: a rapid, sensitive and accurate tool for mapping RNA-seq reads to transcriptomes,” *Bioinformatics*, vol. 32, no. 12, pp. i192–i200, Jun. 2016.
[49] B. Li, V. Ruotti, R. M. Stewart, J. A. Thomson, and C. N. Dewey, “RNA-Seq gene expression estimation with read mapping uncertainty.,” *Bioinformatics*, vol. 26, no. 4, pp. 493–500, Feb. 2010.
[50] L. Pachter, “Models for transcript quantification from RNA-Seq,” Apr. 2011.
[51] J. Foulds, L. Boyles, C. DuBois, P. Smyth, and M. Welling, *Stochastic Collapsed Variational Bayesian Inference for Latent Dirichlet Allocation*. .
[52] J. Lin, “Divergence measures based on the Shannon entropy,” *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, 1991.
[53] U. Syed and G. Yona, “Enzyme Function Prediction with Interpretable Models,” Humana Press, 2009, pp. 373–420.
[54] Illumina, “Considerations for RNA-Seq read length and coverage.” [Online]. Available: http://emea.support.illumina.com/bulletins/2017/04/considerations-for-rna-seq-read-length-and-coverage-.html. [Accessed: 18-Jun-2019].
[55] S. Chhangawala, G. Rudy, C. E. Mason, and J. A. Rosenfeld, “The impact of read length on quantification of differentially expressed genes and splice junction detection,” *Genome Biol.*, vol. 16, no. 1, p. 131, Dec. 2015.
[56] “Coverage and Read Depth Recommendations for Next-Generation Sequencing Applications.” [Online]. Available: https://genohub.com/recommended-sequencing-coverage-by-application/. [Accessed: 18-Jun-2019].
[57] Y. Liu, J. Zhou, and K. P. White, “RNA-seq differential expression studies: More sequence or more replication?,” *Bioinformatics*, vol. 30, no. 3, pp. 301–304, 2014.
[58] Y. Liu *et al.*, “Evaluating the Impact of Sequencing Depth on Transcriptome Profiling in Human Adipose,” *PLoS One*, vol. 8, no. 6, p. e66883, Jun. 2013.
[59] E. S. Lander and M. S. Waterman, “Genomic mapping by fingerprinting random clones: A mathematical analysis,” *Genomics*, vol. 2, no. 3, pp. 231–239, Apr. 1988.
[60] N. J. Schurch *et al.*, “How many biological replicates are needed in an RNA-seq experiment and which differential expression tool should you use?,” *Rna*, vol. 22, no. 6, pp. 839–851, 2016.
[61] M. I. Love, C. Soneson, and R. Patro, “Swimming downstream: statistical analysis of differential transcript usage following Salmon quantification,” *F1000Research*, vol. 7, p. 952, 2018.
[62] G. Giudice, F. Sánchez-Cabo, C. Torroja, and E. Lara-Pezzi, “ATtRACT-a database of RNA-binding proteins and associated motifs,” *Database*, vol. 2016, no. 2, pp. 1–9, 2016.